RESIDUAL RE-ENTRY DEPENDENCE: WHEN SUSPENDED STATES ARE NOT SUFFICIENT FOR RESUMPTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Task interruptions are increasingly common in language-model reasoning, but it remains unclear what information is sufficient for a model to resume a suspended computation. A natural assumption is that once the suspended task state and the next operation are known, the continuation is determined. We test this assumption and find that it can fail: even when the suspended state and continuation operation are held fixed, changing the intervening computation can systematically change the resumed result. We call this remaining influence *residual re-entry dependence*. To measure it, we construct controlled task-switching experiments that hold the suspended state and continuation operation fixed while varying the intervening computation. In a controlled arithmetic task, inserting an intervening computation reduces state-report accuracy from 100% to 76.67% and continuation accuracy from 100% to 72.85%. The two failures are not the same: in 17.99% of interrupted trials the model reports the suspended state correctly but continues incorrectly, while in 14.17% it reports the state incorrectly but nevertheless continues correctly. In a strict within-example factorial, simply changing the relation between the intervening and resumed operators changes continuation accuracy: exact operator matching yields a $+6.08$ percentage-point contrast, with the same direction on Llama-3.1-8B ($+4.95$ pp) but the opposite direction on Qwen2.5-3B (-10.42 pp). Thus, knowing the suspended state alone does not explain how the model resumes. Matched controls rule out two simple explanations: making the intervening numerical state conflict with the suspended state does not consistently worsen continuation, and explicit task-closure cues perform no better than matched filler text. We also find that interruptions broadly change how the resumed computation attends to the suspended-state span, but late-layer activation-patching effects that appear to transfer the state are largely explained by answer identity under same-answer controls. The same pattern appears in a natural reasoning task. On GSM8K, inserting an unrelated factual question changes final-answer accuracy little, whereas inserting another math problem reduces continuation accuracy by 3.2–6.1 percentage points in five of six models; the corresponding state probe degrades in all six. By contrast, in a symbolic permutation task, an apparent interruption effect disappears once operation priming is matched. Together, these results show that the suspended task state is not always a sufficient statistic for resumption: even when the state and next operation are fixed, what the model computes during the interruption can still change what happens after resumption.

1 INTRODUCTION

Language models increasingly operate in settings where one computation is paused, another task intervenes, and the original task must later be resumed. Conversational task switches are known to degrade language-model performance (Gupta et al., 2024), and recent evaluations of reasoning models show substantial fragility under interruptions and dynamic context (Wu et al., 2025). These findings establish that interruptions can matter, but leave a more basic question unresolved: *what does it mean for a model to successfully re-enter a suspended computation?*

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.

A sharper version of this assumption is a sufficiency claim. Let S_A denote the state at which Task A is paused and let O_A denote the operation requested after resumption. If (S_A, O_A) is a sufficient statistic for re-entry, then two histories with the same pair should produce the same resumed result, regardless of the computation performed during the interruption. This is stronger than saying that the model can remember a value: it says that the interruption trajectory has no remaining predictive influence once the suspended state and next operation are fixed.

This claim is experimentally meaningful because it can fail without requiring access to a privileged internal memory variable. We can construct paired histories that share the same symbolic S_A, the same resumed operation O_A, and the same scoring rule, while changing only the intervening trajectory. A systematic difference between the paired continuation outcomes is the phenomenon we call *residual re-entry dependence*. Direct state reports are useful as a second probe, but they are not ground truth for a unique latent register: a model may report a state without using it, or use it without answering a particular report query correctly.

We introduce a controlled study of *reasoning re-entry*: the transition from a suspended task back into computation after an intervening task. The primary estimand is residual re-entry dependence, measured by changing the interruption trajectory while holding (S_A, O_A) fixed. The state report and the continuation are complementary diagnostics: the state report asks whether a direct state query is answered correctly, while the continuation asks whether the state supports the next computation.

The paper follows a falsification-first progression. We test the sufficiency hypothesis in a calibrated arithmetic register task using a strict within-example factorial, branch the same resumed histories into report and continuation probes, test state overwrite, closure, and localized transfer with progressively stronger controls, and finally use a priming-matched symbolic permutation task to delimit the scope of the effect.

Our contributions are fourfold. First, we turn the sufficiency assumption into a paired behavioral test and identify residual re-entry dependence when (S_A, O_A) is fixed. Second, we show that a direct state report is neither necessary nor sufficient for correct continuation. Third, we show that large late activation-patching effects collapse to answer identity under same-answer controls, limiting the claim of localized state transfer. Fourth, we delimit the phenomenon with priming-matched symbolic controls rather than treating every interruption effect as a universal one.

2 Related Work

Interruption, task switching, and context. Conversational task switches and interruptions degrade language-model performance, and reasoning models are fragile under dynamic context (Gupta et al., 2024; Wu et al., 2025). Unnecessary context degrades reasoning (Shi et al., 2023), where relevant information sits matters (Liu et al., 2024), and accuracy falls as input length grows (Levy et al., 2024). Task switching has an independent psychological literature (Monsell, 2003), and interference between successively learned computations is a classical connectionist failure mode (French, 1999; Kirkpatrick et al., 2017). That line asks whether an interruption hurts; we ask whether the interruption trajectory still matters once the suspended state and next operation are fixed.

Traces, faithfulness, and hidden states. Chain-of-thought and scratchpad prompting expose intermediate computation (Wei et al., 2022; Kojima et al., 2022; Nye et al., 2021), and decomposition or resampling strategies build on that surface (Zhou et al., 2023; Wang et al., 2023; Yao et al., 2023), with math benchmarks making it measurable (Cobbe et al., 2021; Suzgun et al., 2022; Lewkowycz et al., 2022). A trace is not a faithful report of the computation behind it (Turpin et al., 2023; Lanham et al., 2023), and step-level analyses ask which steps carry the work (Mehrafarin et al., 2026; Bogdan et al., 2025). Hidden-state work reads activations directly: feed-forward layers act as key–value memories (Geva et al., 2021), probing recovers syntactic and factual structure (Belinkov, 2022; Hewitt and Manning, 2019; Jiang et al., 2020; Petroni et al., 2019), circuits are traced through attention and the residual stream (Vaswani et al., 2017; Elhage et al., 2021; Olsson et al., 2022; Wang et al., 2023), and in-context learning has been described as building a task vector in activation space (Hendel et al., 2023; Todd et al., 2024). Attention is the usual target for such claims, but few heads carry much of the work (Clark et al., 2019; Voita et al., 2019), attention flow decomposes across heads and layers (Abnar and Zuidema, 2020), and weights are contested as explanations (Jain and Wallace, 2019; Wiegreffe and Pinter, 2019). We use such probes but not as ground truth: our state report is one

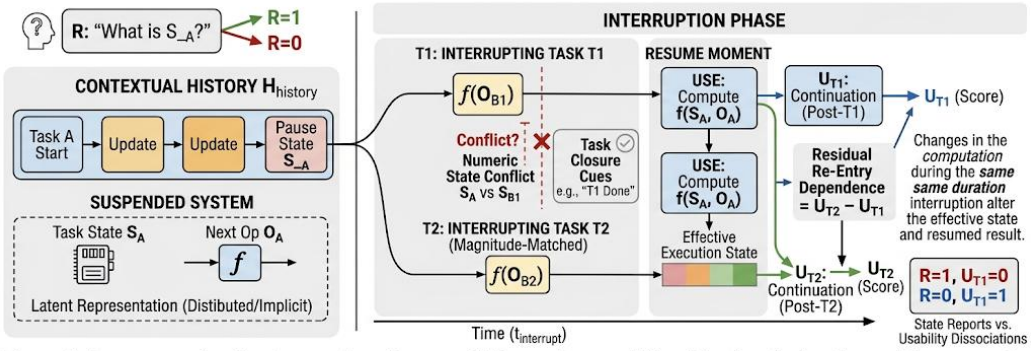


Figure 1: Residual re-entry dependence and the report–continuation dissociation. The suspended Task-A state and resumed operation are held fixed while the intervening computation changes the re-entry context. The state report (R) and the continuation (U) are separate probes of the same resumed history.

behavioral probe among two, and we report the attention shift as a description of what the resumed computation reads.

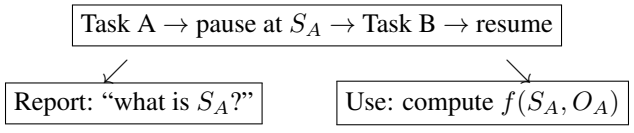
Why a probe is not a readout. Probe accuracy can reflect task-specific shortcuts rather than the structure of interest (Ravfogel et al., 2020; Elazar and Goldberg, 2018), verbalized confidence is miscalibrated (Jiang et al., 2021), and behavioral tests can fail to isolate the intended variable (Finlayson et al., 2021). Models do carry usable internal signals about what they know (Kadavath et al., 2022), and uncertainty can be estimated from meaning rather than surface form (Kuhn et al., 2023; Farquhar et al., 2024). Numerical competence is itself sensitive to elicitation (Wallace et al., 2019; Stolfo et al., 2023). These findings motivate our two-probe design, in which a correct report is neither necessary nor sufficient for a correct continuation.

Causal intervention and its interpretation. Activation patching is causal, but its interpretation depends on patch location, metric, and donor construction (Zhang and Nanda, 2024; Heimersheim and Nanda, 2024; Ghandeharioun et al., 2024), and apparently meaningful patched subspaces admit alternative causal explanations (Makelov et al., 2024). Related work localizes and rewrites specific computations (Meng et al., 2022; Meng et al., 2023; De Cao et al., 2021), traces how computation changes over training (Nanda et al., 2023), and automates the search for the responsible subgraph (Conmy et al., 2023). We treat a large patch effect as a hypothesis to falsify and report where it collapses.

Interactive and augmented computation. ReAct ties actions to a changing textual computation state (Yao et al., 2023); models are now equipped with tools, retrieval, and self-critique (Schick et al., 2023; Mialon et al., 2023; Shinn et al., 2023), persistent agentic state (Park et al., 2023), or an interpreter that externalizes arithmetic (Gao et al., 2023; Chen et al., 2023). Adaptation of a frozen pretrained model to instruction following underlies all of these (Brown et al., 2020; Touvron et al., 2023; Ouyang et al., 2022). Our controlled tasks instead isolate re-entry with known state and answer, separating the residual dependence, the state report, and the continuation before open-world interaction.

3 Problem Setup: When Is a Suspended State Sufficient?

Let a Task-A trajectory be suspended after reaching a behaviorally defined state S_A. An intervening Task B is then executed, after which the model receives a resume cue. From the same resumed history H, we evaluate two branches:



We denote report correctness by $\mathrm{R} \in \{0, 1\}$ and continuation correctness by $\mathrm{U} \in \{0, 1\}$. A report-use dissociation includes both $\Pr(\mathrm{R} = 1, \mathrm{U} = 0)$ and $\Pr(\mathrm{R} = 0, \mathrm{U} = 1)$. The first shows that a correct report is not sufficient for computational use; the second shows that a failed report is not sufficient evidence that the relevant computation is unavailable.

For two interruption trajectories I and I', define the paired residual re-entry contrast

$$\Delta_{\mathrm{RRE}}(I, I'; S_A, O_A) = \Pr(U = 1 \mid S_A, O_A, I) - \Pr(U = 1 \mid S_A, O_A, I').$$

Under the sufficient-statistic hypothesis, this contrast should be zero up to sampling variation. A nonzero contrast does not identify a single mechanism; it establishes that the observable resumed computation depends on more than (S_A, O_A). We call this remaining influence *residual re-entry dependence*. It may reflect state competition, operator carryover, query-conditioned reconstruction, or another interaction between the suspended and intervening trajectories.

The distinction is easiest to see as a measurement problem. A state query asks the model to produce a token sequence that names a value; a continuation asks it to transform the value under a new operator. These probes share context but differ in the computation demanded after the resume cue. Consequently, the state report is not a privileged readout of a latent state, and the continuation is not simply the state report followed by a deterministic arithmetic step. We use the two probes as complementary behavioral tests and interpret their disagreement as evidence about the re-entry interface, not as proof that one probe is the "true" measure of memory.

For a condition c, we summarize the two probes by

$$\mathrm{Rep}(c) = \mathbb{E}[\mathrm{R} \mid c], \qquad \mathrm{Use}(c) = \mathbb{E}[\mathrm{U} \mid c],$$

and by the off-diagonal rates $\mathbb{E}[\mathrm{R}(1 - \mathrm{U}) \mid c]$ and $\mathbb{E}[(1 - \mathrm{R})\mathrm{U} \mid c]$. In paired experiments, the same base example and the same assigned state are retained across conditions whenever the protocol permits. This is important because a high aggregate continuation rate can coexist with a low report rate for one subset of examples, while averaging can hide the complementary pattern on another subset.

4 Experimental Design: Testing Residual Dependence

4.1 RegisterOps: calibrated arithmetic re-entry

RegisterOps contains independent Task-A and Task-B integer registers. Task A is updated and paused; Task B is then executed; Task A is resumed. All principal report and continuation probes fork from the identical resumed conversation. Clean controls use the same pause/resume protocol without an intervening Task-B computation.

For the strict operator experiment, both tasks use operators in $\{+1, +2, +3, -1, -2, -3\}$. Each suspended-A base history is reused across all six Task-B operators, and each resulting history across all six resumed-A operators, so every 6×6 comparison is within base example rather than across independently sampled problems; we cluster bootstrap by base identity.

4.2 Mechanistic falsification with answer-collision donors

We treat a large patch effect as a hypothesis to falsify, not as sufficient evidence that a named reasoning variable has been localized.

We construct answer-collision triples. A target pair (S_t, O_t) produces answer Z. A *state donor* (S_d, O_t) and an *operator donor* (S_t, O_d) are chosen so that both donors produce the same alternative answer $Y \neq Z$. We patch a single residual-stream position at four temporal anchors: `resume_end`, `instruction_end`, `assistant_frontier`, and `answer_frontier`. If structurally different donors acquire the same effect only at the final anchor, the parsimonious explanation is answer-level transfer. We also replay each recorded conversation with and without the intervening turn, comparing the attention distribution of the answer position, which measures the re-entry computation directly.

4.3 SHUFFLEOPS: SYMBOLIC RE-ENTRY WITH MATCHED PRIMING

ShuffleOps tracks symbolic objects across four slots under swap operations. An initial version suggested a large symbolic interruption effect, but additional controls revealed that simply executing a recent swap strongly primes later swap execution. We therefore use a stricter v9.1 design with four matched conditions. In `same_A`, `cross_B`, and `cross_C` the prime segment applies the same swap twice, drawn from the Task-A, Task-B, or Task-C object set; in `neutral` the same two turn positions record a swap label without applying it. Because a swap is an involution, $T_p(T_p(S)) = S$, so turn count, swap count, prime operator, continuation operator, and the final Task-A state are matched across conditions, and the same-task/cross-task contrast isolates task-set membership from recency of swap execution.

A priming diagnostic repeats the probe after one of four histories—no recent swap, a generic third-set swap, the pair's own Task-B swap, or an extra Task-A swap—separating recency of swap execution from task-set membership.

The preregistered positive criterion was not met, and a downstream state-verification probe itself varies strongly by condition, so conditioning on verification success breaks the paired sample. We therefore use the common-prefix sample as an intent-to-treat-style descriptive analysis, report verification-conditioned complete cases only as sensitivity analysis, and do *not* treat ShuffleOps as evidence for a universal positive interruption effect (Appendix R).

5 PROTOCOL AND ESTIMANDS

5.1 HOLDING THE STATE FIXED WHILE MOVING ONLY THE INTERRUPTION

The central design constraint is that report and use are evaluated on the same resumed history rather than on two independently sampled prompts. We first construct Task A, execute the prescribed update, and stop at a known state. The intervening segment is then rendered with a fixed number of turns and a fixed response format. After the resume cue, we clone the resulting conversation into separate branches. One branch asks for a direct state report; the other requests the next Task-A operation. No response from one branch is shown to the other. This makes the two outcomes comparable while preserving the fact that the query itself can alter computation.

For every trial we retain the prompt, ground-truth state, expected continuation, readout wording, raw response, and parser flags. A trial is not silently removed for being verbose: protocol validity, report correctness, continuation correctness, and parse success are separate variables (Appendices B and E). This matters in v9.1, where a verification probe has condition-dependent success and cannot be a neutral data-quality filter.

5.2 SEPARATING RESIDUAL DEPENDENCE FROM PROBE ARTIFACTS

Let R_i indicate an exact state report and U_i indicate a correct continuation for trial i. We report the marginal state-report rate $\hat{r} = \frac{1}{n}\sum_i R_i$ and continuation rate $\hat{u} = \frac{1}{n}\sum_i U_i$, together with the two off-diagonal frequencies

$$d_{R\to U} = \Pr(R = 1, U = 0), \qquad d_{U\to R} = \Pr(R = 0, U = 1).$$

The first is the main RegisterOps dissociation: it tests directly whether a correct report is sufficient for resumed computation. In the symbolic experiment we additionally report the difference between a same-task prime and the mean of two cross-task primes,

$$\Delta_{\text{switch}} = \text{Acc}(\texttt{same_A}) - \tfrac{1}{2}\{\text{Acc}(\texttt{cross_B}) + \text{Acc}(\texttt{cross_C})\}.$$

This estimand is descriptive after the preregistered gate failed; it is not relabeled as a causal interruption effect.

5.3 DECIDING WHAT EACH CONTRAST IS ALLOWED TO CLAIM

RegisterOps operator effects are computed within a base example. We resample base identities, retain all factorial cells belonging to each selected base, and recompute the contrast. This avoids treating the

Table 1: Summary of the controlled contrasts. Each is paired or taken within base example; intervals are cluster bootstrap over base histories and p-values are exact McNemar tests. Per-condition tables: Tables 13, 14, 10, and the appendix.

Contrast	Models	Effect	Interval / p
RegisterOps, clean→interrupted: report / continuation	Qwen2.5-7B	-23.3 / -27.2 pp	
Operator factorial, exact vs. unmatched	7B / Llama-3.1-8B / 3B	$+6.08$ / $+4.95$ / -10.42 pp	CIs in App.
Attention on the suspended state, interrupted−clean	7B / 3B / Llama-3.1-8B	-0.029 / -0.051 / -0.021	CI < 0
State congruency, conflict vs. none	Qwen2.5-7B → Qwen3-4B	-0.10 to -10.00 pp	
GSM8K, math interruption / state probe	6 models	-3.2 to -6.1 / -1.7 to -33.0 pp	$5/6$, $6/6$ sig.

6×6 cells that share a history as independent observations. For v8, the layer used for confirmation is fixed before reading the confirmation split. For v9.1, we report both the common-prefix sample and the verification-conditioned complete cases. The common-prefix analysis preserves the assigned condition even when a downstream readout fails; the complete-case analysis answers a different, conditional question.

Gates are reported rather than tuned after observing the result. v9.1 requires a calibration and continuation threshold that no model in the final matched runs satisfies, so we do not call any v9.1 model a successful symbolic re-entry system. Calibration counts are low: Qwen2.5-7B selected its wording from 1 valid item out of 108, 14B from 14/108, and 32B from 23/108. These counts limit the symbolic cross-model comparison rather than license discarding unfavorable conditions.

6 Results: Residual Dependence and Its Boundaries

Table 1 collects the contrasts reported in this section; the subsections below give the design of each experiment and the full tables are in the appendix. The appendix gives the provenance of every number (A), the protocol rules (D–G), the full factorial and contrasts (H–N), the attention analysis by layer and head (O), a recorded GSM8K failure (P), the symbolic sensitivity tables (Q–T), and the patch controls (U–W).

6.1 Fixed state and operation do not fix the result

The strict paired RegisterOps result for Qwen2.5-7B-Instruct (Qwen Team et al., 2024) first tests the sufficiency hypothesis (Table 8). Clean report and continuation are both 100%; after an intervening computation they fall to 76.67% and 72.85%. The two off-diagonal cells dissociate in both directions on the same resumed histories: 17.99% of trials have $\mathrm{R} = 1, \mathrm{U} = 0$, so a correct report is not sufficient for computational use, and 14.17% have $\mathrm{R} = 0, \mathrm{U} = 1$, so it is not necessary either (Appendices J and I).

The result is the first direct evidence for residual re-entry dependence: the interruption is not behaviorally irrelevant after the suspended state and resumed operation are specified. It does not prove a unique latent scalar S_A, nor identify whether the residual influence is carried by a state representation, an operation trace, or the re-entry instruction.

6.2 Residual dependence is structured by operator relation

In the strict within-example 6×6 factorial, the residual contrast depends systematically on the relation between the intervening and resumed operators. The exact-match comparison estimates a $+6.08$ pp change in continuation accuracy (95% bootstrap CI $[1.67, 10.58]$), which is the main empirical estimate of Δ_{RRE} (Figure 2). Matching operator magnitude improves accuracy by an additional 4.17 pp (CI $[1.25, 7.08]$), whereas matching only the arithmetic sign yields $+0.97$ pp with a confidence interval crossing zero.

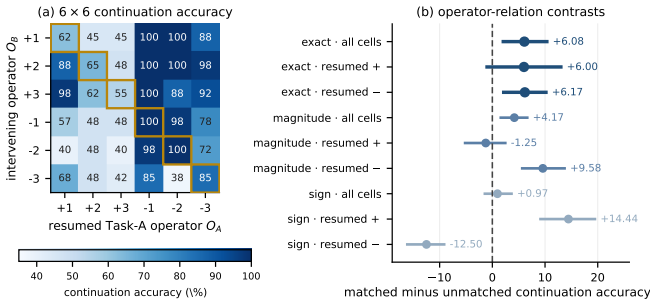


Figure 2: RegisterOps operator factorial. (a) Continuation accuracy for each intervening/resumed operator pair, computed within base example; the gold outline marks the exact-match diagonal. (b) Matched-minus-unmatched continuation accuracy for the exact, magnitude, and sign relations, overall and stratified by the sign of the resumed operator, with cluster-bootstrap 95% intervals over base histories. Exact matching helps in both strata, whereas the near-zero aggregate sign-match estimate cancels two significant effects of opposite direction.

Because the resumed operator also changes task difficulty, we check that the exact-match result is not an artifact of operator easiness. Resumed additions are substantially harder than resumed subtractions (55.8% versus 89.9% continuation accuracy), but the matched-versus-unmatched exact-operator contrast is stable within both strata: $+6.00$ pp when the resumed operator is an addition and $+6.17$ pp when it is a subtraction. The other two contrasts are not uniform: magnitude matching is worth $+9.58$ pp for resumed subtractions but -1.25 pp for resumed additions, and sign matching is $+14.44$ pp for resumed additions and -12.50 pp for resumed subtractions. The near-zero aggregate sign-match estimate is therefore a cancellation of two opposite significant effects.

The same factorial was run on two further models under the same protocol and 40-base design. The exact-operator contrast is $+4.95$ pp (CI $[3.83, 6.08]$) for Llama-3.1-8B, consistent across the discovery and confirmation splits, and -10.42 pp (CI $[-13.00, -8.17]$) for Qwen2.5-3B, so re-entry structured by the operator relation replicates on a second model family and reverses on a third. Only the continuation branch of these runs is usable: their clean state probe was never recorded and their interrupted probe never returns the paused value.

The error structure is compatible with operator-related interference but not with complete operator substitution. Of 391 continuation errors, 85 equal the result of applying the full Task-B operator to the correct paused A state, and 29 retain the Task-A sign while using the Task-B magnitude. These two categories account for roughly 29.2% of errors. We therefore describe the result as *structured dependence on the operator relation* rather than a universal carryover mechanism.

6.3 Residual dependence is not explained by state conflict or closure

Two initially plausible explanations do not survive matched controls. Making the Task-B numeric state conflict with the Task-A suspended state does not reliably hurt more than a congruent state, and direct capture of the B value is rare. Semantic closure cues such as "Task B is complete" do not outperform a matched filler turn either. In the v6.1 paired comparison over 720 examples, closure text improves continuation over an open prompt by 4.58 pp but is indistinguishable from a matched filler turn (0.00 pp, CI $[-1.53, 1.67]$), whereas explicitly externalizing B improves continuation by 8.89 pp but is statistically indistinguishable from a matched unrelated-number turn ($+0.69$ pp, CI crossing zero). The benefit therefore tracks recency and externalization of the intervening value rather than semantic closure, so neither explanation accounts for the main re-entry failures (Appendices K and L).

A separate multi-model experiment tests the conflict question at larger scale on a six-operation register task with 2000 bases per condition. Across five model runs, a congruent intervening state and a conflicting one differ by at most 1.0 pp: what changes continuation is whether the intervening segment updates the same register at all, not whether its final value collides with the suspended state. The conflict-minus-none contrast ranges from -0.10 pp for Qwen2.5-7B to -10.00 pp for Qwen3-4B, the unrelated task does not hurt Llama-3.1-8B, and direct capture of the intervening value

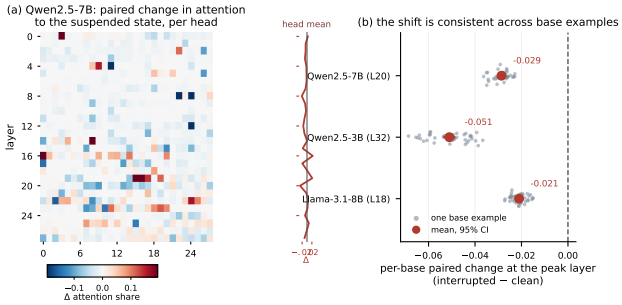


Figure 3: Attention re-entry analysis. Each recorded trial is replayed as produced and again with the intervening Task-B turn deleted, and the attention distribution of the final answer position is read out. (a) Paired change in the share of that attention landing on the span holding the suspended Task-A value, per layer and head, for Qwen2.5-7B; the right strip is the head-averaged layer profile with cluster-bootstrap 95% intervals over base histories. (b) Per-base paired change at each model's peak layer: dots are the 40 base examples and the bar is the mean with its cluster-bootstrap interval. The direction of the shift replicates across the three models, but its magnitude and peak layer do not.

stays below 2.5% of trials. The direction is therefore model-dependent rather than universal; Table 13 gives the full matrix, including two independent data seeds and the two models that fail the protocol.

Prompt wording can nevertheless change what state is reported. An earlier restoration cue caused a systematic rollback to Task A's initial value rather than its most recently computed value, in 14.4% of readout trials. This motivates treating state reports as instruction-conditioned behaviors rather than transparent reads of a fixed memory slot.

6.4 The interruption reallocates attention at re-entry

Because the patch assay cannot test localization at the early anchors, we measure the re-entry computation directly. We replay each recorded conversation twice, once as the model produced it and once with the intervening Task-B turn deleted, append the same continuation request, and read the attention distribution of the final answer position. The clean branch is the counterfactual exactly: same base, same suspended state, same resumed operator, and only the intervening segment differs.

Figure 3(a) reports the paired change in attention to the span carrying the suspended value, by layer and head. The reallocation is broad rather than concentrated in a few heads: 764 of 784 layer–head cells have a cluster-bootstrap interval excluding zero, 572 negative and 192 positive, and the head-averaged share falls in 20 of 28 layers and rises in 8 (Appendix O). The largest single reduction is at layer 20 (Table 14). The intervening Task-B turn itself retains 0.067 of the answer position's attention, so the interruption rewrites which heads read the suspended state rather than switching attention off. Figure 3(b) shows every base example at each model's peak layer: the direction replicates across model families, although the magnitude and the peak layer do not.

Two limits apply. Because the shift is a reallocation rather than a monotone loss, it does not map onto errors: within interrupted trials, state-span attention at layer 20 is slightly *higher* on incorrect continuations (0.0758) than on correct ones (0.0692). The measurement is also correlational: it shows that (S_A, O_A) does not fix the computation that follows, without tracing the causal path (Appendix V).

6.5 Natural-task re-entry depends on what the interruption computes

The controlled tasks above use synthetic registers, so we ran the same paradigm on GSM8K: the model sees a reference solution up to its midpoint, an interruption is inserted, and it must finish and emit the final answer, scored by exact match. Six models ran 1000 problems per condition, with either an unrelated factual question or a second GSM8K problem as the interrupter (protocol in Appendix F).

An unrelated factual question is close to harmless: the paired change lies between -2.2 and $+0.9$ pp and reaches significance only for Qwen2-7B (-2.2 pp, $p = 0.010$). A second math problem degrades the continuation in five of the six models, by -3.2, -3.2, -3.4, -5.5 and -6.1 pp (all $p < 0.005$);

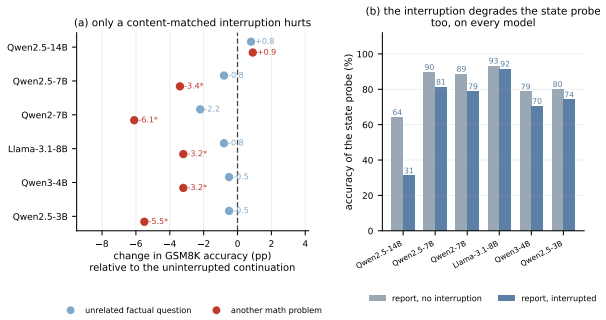


Figure 4: Interrupted chain-of-thought on GSM8K, 1000 problems per condition and six models, exact-match scoring. (a) Paired change in final-answer accuracy relative to the uninterrupted continuation; stars mark McNemar $p < 0.05$. An unrelated factual interruption is close to harmless, whereas a second math problem degrades the continuation in five of six models. (b) The same interruption also degrades a direct probe for the last computed value, by up to 33 points.

the exception is Qwen2.5-14B, whose final answer is unaffected ($+0.9$ pp). What the interruption computes therefore matters, which a sufficient-statistic account of the suspended state forbids. The two interruptions differ in whether they draw on the same computation the resume requires, and this is the condition that separates successful from unsuccessful re-entry. The state probe answers differently again: it degrades under interruption in all six models, from -1.7 pp for Llama-3.1-8B to -33.0 pp for Qwen2.5-14B (all $p < 0.001$). The largest model is thus the one whose final answer is most robust and whose state probe is most fragile, the report–continuation dissociation at natural-task scale.

7 Discussion

The central result is a failure of sufficiency, not a claim that models forget a scalar register: with the suspended state and resumed operation fixed, changing the interruption trajectory still changes both the resumed result and the attention the resumed computation reads. The report–continuation split shows why direct state probes are not enough: naming the state does not guarantee using it, and failing to name it does not establish that it is unusable.

The controls narrow the mechanisms rather than eliminate them: state conflict and closure do not explain the effect, same-answer donors show the late patch effect tracks answer identity, and priming removes the earlier symbolic signal. The result is a task-specific and model-dependent residual dependence rather than a universal penalty. No single probe identifies a unique internal variable called "the suspended state"; the probes are informative because they constrain one another.

8 Limitations

The strongest positive result comes from controlled arithmetic tasks and is model-dependent: the exact-operator effect reverses on Qwen2.5-3B, the state-conflict contrast spans -0.10 to -10.00 pp, and numerical competence is probe-sensitive (Wallace et al., 2019). Symbolic controls do not reproduce a universal interruption effect, state-report calibration makes their report and continuation values diagnostic, and the patch result covers only single-position interventions. Two conflict models fail the side-task protocol and the six-operation task sits near a capability boundary (Appendix W).

9 Conclusion

Holding the suspended state and resumed operation fixed does not remove residual re-entry dependence: the state report and the continuation dissociate on synthetic and natural tasks alike, so the two probes should be measured separately.

Reproducibility Statement

All controlled tasks are generated programmatically with recorded seeds and explicit ground-truth states. Principal RegisterOps operator effects use a within-base factorial with cluster bootstrap by base identity; mechanistic layer selection uses a preassigned discovery/confirmation split. ShuffleOps reports the preregistered gate failure, the common-prefix paired analysis, and verification-conditioned complete-case sensitivity rather than only the favorable filter. Code, run manifests, prompt templates, and processed result tables will be included in the supplementary material.

AI Use Statement

We used generative AI tools for research-idea and experimental-design discussion, prose drafting and editing, code drafting and debugging, experiment organization, literature search, and preparation of tables and figures. We did not use them to generate experimental runs, alter empirical values, select favorable results after observing outcomes, or replace author verification of scientific claims. Authors reviewed all executions, statistical outputs, interpretations, citation metadata, and final text, retained AI suggestions only when the recorded evidence supported them, and take responsibility for the final content.

References

Akash Gupta, Ivaxi Sheth, Vyas Raina, Mark Gales, and Mario Fritz. LLM task interference: An initial study on the impact of task-switch in conversational history. In *EMNLP*, pages 14633–14652, 2024.

Tsung-Han Wu, Mihran Miroyan, David M. Chan, Trevor Darrell, Narges Norouzi, and Joseph E. Gonzalez. Are large reasoning models interruptible? arXiv:2510.11713, 2025.

Asma Ghandeharioun, Avi Caciularu, Adam Pearce, Lucas Dixon, and Mor Geva. Patchscopes: A unifying framework for inspecting hidden representations of language models. In *ICML*, volume 235 of PMLR, pages 15466–15490, 2024.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *ICLR*, 2024.

Aleksandar Makelov, Georg Lange, Atticus Geiger, and Neel Nanda. Is this the subspace you are looking for? An interpretability illusion for subspace activation patching. In *ICLR*, 2024.

Stefan Heimersheim and Neel Nanda. How to use and interpret activation patching. arXiv:2404.15255, 2024.

Houman Mehrafarin, Amit Parekh, and Ioannis Konstas. When chain-of-thought fails, the solution hides in the hidden states. In *Findings of EMNLP*, 2026. arXiv:2604.23351.

Paul C. Bogdan, Uzay Macar, Neel Nanda, and Arthur Conmy. Thought anchors: Which LLM reasoning steps matter? arXiv:2506.19143, 2025.

Qwen Team et al. Qwen2.5 technical report. arXiv:2412.15115, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *NeurIPS*, 2022.

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

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *NeurIPS*, 2022.

Nicola De Cao, Wilker Aziz, and Ivan Titov. Editing factual knowledge in language models. In *EMNLP*, 2021.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *EMNLP*, pages 5484–5495, 2021.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438, 2020.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection. In *ACL*, 2020.

Yanai Elazar and Yoav Goldberg. Adversarial removal of demographic attributes from text data. In *EMNLP*, 2018.

John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In *NAACL*, 2019.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 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, 2021.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *EMNLP-IJCNLP*, pages 2463–2473, 2019.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. Do NLP models know numbers? Probing numeracy in embeddings. In *EMNLP-IJCNLP*, pages 5307–5315, 2019.

Matthew Finlayson, Aaron Mueller, Sebastian Gehrmann, Stuart M. Shieber, Tal Linzen, and Yonatan Belinkov. Causal analysis of syntactic agreement mechanisms in neural language models. In *ACL-IJCNLP*, pages 1828–1843, 2021.

Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis. In *EMNLP*, 2023.

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *ACL*, pages 4190–4197, 2020.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *TMLR*, 2023.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does BERT look at? An analysis of BERT's attention. In *BlackboxNLP Workshop*, pages 276–286, 2019.

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:2110.14168, 2021.

Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. In *NeurIPS*, 2023.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021.

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

Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. PAL: Program-aided language models. In *ICML*, pages 10764–10799, 2023.

Roee Hendel, Mor Geva, and Amir Globerson. In-context learning creates task vectors. In *Findings of EMNLP*, pages 9318–9333, 2023.

Sarthak Jain and Byron C. Wallace. Attention is not explanation. In *NAACL*, pages 3543–3556, 2019.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. arXiv:2207.05221, 2022.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *PNAS*, 114(13):3521–3526, 2017.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *ICLR*, 2023.

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:2307.13702, 2023.

Mosh Levy, Alon Jacoby, and Yoav Goldberg. Same task, more tokens: The impact of input length on the reasoning performance of large language models. In *ACL*, pages 15339–15353, 2024.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. In *NeurIPS*, 2022.

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. *TACL*, 12:157–173, 2024.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *ICLR*, 2023.

Grégoire Mialon, Roberto Dessì, Maria Lomeli, Christoforos Nalmpantis, Ramakanth Pasunuru, Roberta Raileanu, Baptiste Rozière, Timo Schick, Jane Dwivedi-Yu, Asli Celikyilmaz, et al. Augmented language models: A survey. *TMLR*, 2023.

Stephen Monsell. Task switching. *Trends in Cognitive Sciences*, 7(3):134–140, 2003.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *ICLR*, 2023.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. Show your work: Scratchpads for intermediate computation with language models. arXiv:2112.00114, 2021.

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.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *UIST*, 2023.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *NeurIPS*, 2023.

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 *ICML*, pages 31210–31227, 2023.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *NeurIPS*, 2023.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. Challenging BIG-Bench tasks and whether chain-of-thought can solve them. arXiv:2210.09261, 2022.

Eric Todd, Millicent L. Li, Arnab Sen Sharma, Aaron Mueller, Byron C. Wallace, and David Bau. Function vectors in large language models. In *ICLR*, 2024.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. LLaMA: Open and efficient foundation language models. arXiv:2302.13971, 2023.

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 *NeurIPS*, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *ACL*, pages 5797–5808, 2019.

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

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: A circuit for indirect object identification in GPT-2 small. In *ICLR*, 2023.

Sarah Wiegreffe and Yuval Pinter. Attention is not not explanation. In *EMNLP*, pages 11–20, 2019.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *NeurIPS*, 2023.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V. Le, and Ed H. Chi. Least-to-most prompting enables complex reasoning in large language models. In *ICLR*, 2023.

A Provenance: from raw runs to reported numbers

Every number in this paper is produced by the same three-stage pipeline, and this section makes the chain explicit so that a reader can retrace any single value from the manuscript back to a recorded trial.

1. **Raw runs.** Each experiment writes one JSONL record per trial: the fully rendered prompt, the model response, the parsed fields, and the protocol flags. Records are never edited after the run.
2. **Analysis.** One script per experiment reads the JSONL records, applies the protocol and parser rules of Section D, and writes a processed CSV with one row per contrast or per condition.

Table 2: Provenance of every reported result. Each row maps a result in the paper to the processed file it is generated from and to the raw run directory that file is derived from. Sizes are the on-disk size of the raw directory.

Reported result	Processed data (`data/`)	Raw runs
Tables 1, 8, 5–7; Fig. 2	`v7_registerops_core.csv` and `_factorial`, `_contrasts`, `_joint`, `v7_cross_model_registerops.csv`	`rsr7_run/reasoning_state_recovery_v7_iclr_core/runs` (11 JSONL, 110 MB)
Table 13	`v2_cross_model_state_congruency.csv`	`rsr_v2_run` (40 JSONL, 406 MB)
Tables 11, 12	`v61_conditions.csv`, `v61_paired_effects.csv`	`rsr6_run/rsr6_v61_branch_verified/runs` (8 JSONL, 17 MB)
Appendix N	`v52_conflict_protocol.csv`	`rsr5_v52_branch_verified/runs` (16 JSONL, 36 MB)
Tables 14, 15; Fig. 3	`v10_attention_*.csv`, `v10_perhead_delta_*.npz`	`rsr10_run/runs` (7 JSONL, 3 NPZ, 42 MB)
Table 10, Table 16; Fig. 4	`v11_gsm8k_interruption.csv`	`rsr11_run/runs` (11 JSONL, 18 MB)
Tables Q, 18; Figs. 5, 6	`v91_*.csv`, `v9_priming_diagnostic.csv`	`rsr91_run/runs`, `rsr9_run/shuffleops/runs` (70 JSONL)
Tables 19, 20; Fig. 7	`v8_patch_delta_target.csv`, `v8_donor_target_cosine.csv`, `v2_*.csv`	`rsr8_run/runs` (66 JSONL, 26 MB), `rsr_v2_run`

Table 3: Controlled experiment inventory. Counts are the recorded raw or common-prefix counts used in the analysis; they are not claims of independently sampled users or tasks.

Experiment	Model / scale	Bases	Analysis scope
RegisterOps	Qwen2.5-7B	40	clean vs. interrupted report and continuation; 6×6 operator factorial
RegisterOps replication	Qwen2.5-3B / Llama-3.1-8B	40 each	6×6 operator factorial, continuation branch only
State congruency	5 models	2000 each	none / congruent / conflict / strong-resume / unrelated
Answer collision	Qwen2.5-7B	$N = 216$	four temporal anchors; state and operator donors
Attention re-entry	Qwen2.5-7B / 3B	40 each	replay of recorded histories; clean vs. interrupted
GSM8K re-entry	Qwen2.5-7B / 3B, Llama-3.1-8B, Qwen3-4B	300 each	clean / unrelated / math interruption; state probe
Priming diagnostic	Qwen2.5-7B / 14B / 32B	24 / 8 / 6	four recent-swap histories before the probe
ShuffleOps	Qwen2.5-7B	24	864 common-prefix cells per condition
ShuffleOps	Qwen2.5-14B	12	432 common-prefix cells per condition
ShuffleOps	Qwen2.5-32B	12	432 common-prefix cells per condition

3. **Reporting.** Every table and figure is generated from those processed CSVs. The manuscript quotes the processed values and nothing else; no number is typed in by hand from a console.

Two consequences are worth stating explicitly. First, the denominators in Table 3 and in every contrast table are the recorded raw counts, not an assumption about independent sampling. Second, where a probe fails for protocol reasons—for example the two state-congruency runs whose side task does not verify—the failure is recorded in the same files as the successes, so the exclusion is auditable rather than implicit.

B EXPERIMENT INVENTORY

C WHAT THE APPENDIX ANSWERS

This appendix is organized around the questions a reviewer can ask after reading the main claim. First, is the fixed-state counterfactual implemented at the level of actual prompts, rather than only in an aggregate table? Section J gives a raw worked example. Second, could the result be explained by a closure cue, an extra turn, or a newly exposed number? Section K gives a matched v6.1 case and Section L the complete condition and paired tables. Third, do the filtering and calibration decisions change the symbolic conclusion? The sensitivity tables report both common-prefix and complete-case denominators, and Section Q reports the condition-level profiles in full. Fourth, what is the strongest mechanistic claim that the patch experiment supports? The interpretation-boundary section states the precise limit, and Section W gives the earlier same-direction patch controls. Fifth, does the phenomenon extend beyond one model? Section M reports the cross-model state-congruency test, including the two models that fail the protocol, and Section N gives an independent branch-verified protocol.

D PROTOCOL AND SCORING DETAILS

The worked RegisterOps row is stored in `rsr7_run/reasoning_state_recovery_v7_iclr_core/runs/qwen25_v7/behavior/paired_rows.jsonl`; the worked v6.1 row is stored in `rsr6_run/reasoning_state_recovery_v6_operator_closure/rsr6_v61_branch_verified/runs/qwen25_v61_formal/raw/factorial_rows.jsonl`. RegisterOps uses independently initialized Task-A and Task-B registers. The state target is defined by the programmatic register value at the pause boundary, not by a model-generated explanation. A report is correct only when the extracted state equals this target; a continuation is correct only when the extracted answer equals the deterministic result of applying the resumed Task-A operator. We preserve protocol and scoring flags separately so that formatting failures are visible in the ledger.

The strict operator analysis uses the six signed increments $\{+1, +2, +3, -1, -2, -3\}$. Exact matching means the signed operators are equal; magnitude matching means their absolute values are equal; sign matching means their signs agree. These indicators are nested, so the reported contrasts should be read as paired descriptive effects for the corresponding relation, not as independent randomized treatments. Confidence intervals are cluster bootstrap intervals over base histories.

For v8, the target and donor prompts are constructed so that the target pair yields answer Z, while a state donor and an operator donor both yield answer Y. The patch metric is the probability assigned to Y at the designated answer readout. The four anchors are chosen from the same tokenized prompt: the end of the resumed history, the end of the next instruction, an assistant-generation frontier, and the answer frontier. The claim is limited to the tested residual-stream intervention.

E RAW PROTOCOL FIELDS AND PARSER RULES

For RegisterOps, each raw row records `base_id`, `a_start`, `a_pre_op`, `pause_state`, `b_start`, `b_op`, `b_final`, `a_op`, `gold_cont`, `readout_pred`, `continuation_pred`, protocol flags, and error taxonomy. The clean branch records the same report and continuation fields without the Task-B segment. A report is correct only when the extracted integer equals `pause_state`; a continuation is correct only when the extracted integer equals `gold_cont`. Rows with malformed protocol responses are retained in the raw ledger and excluded only from the estimand for which the relevant validity gate is required.

For v9.1, each row records the complete initial arrangements, the pre-swap, prime condition, prime verification flags, readout gold/prediction, continuation gold/prediction, relation label, and selected wording. This makes it possible to audit whether `same_A`, `cross_B`, `cross_C`, and `neutral` differ in final state, swap count, or verification coverage. The complete-case analysis is explicitly secondary because `prime_ok` is downstream of the assigned condition.

F NATURAL-TASK PROTOCOL AND EARLIER RUNS

Section 6.5 reports a controlled GSM8K re-entry experiment. Each problem is cut at its midpoint, the reference steps are shown as a history, an interruption is inserted together with a restoration cue, and the model must emit the final answer in a fixed format, which we score by exact match on the token after `####`. The two interruption conditions use either an unrelated factual question or the first steps of a different GSM8K problem; the state probe asks for the last computed value in the history and is scored against the value in the final shown step. Every condition was run on the same 1000 problems for each of six models, so all reported comparisons are paired and tested with exact McNemar tests.

Table 10 gives the per-condition accuracies behind Figure 4.

The protocol was chosen after an audit of earlier GSM8K and BBH runs. Those runs used heterogeneous output formats and, in some conditions, permissive numeric matching that counts any number equal to the gold value as correct; the loose and strict scorers differ by $21.7\,\mathrm{pp}$ on the main interruption contrast and by $29.0\,\mathrm{pp}$ on the strong-prompt contrast. We therefore do not use those numbers as cross-model evidence. The same audit also found that two files labelled as Llama baselines are byte-identical to the DeepSeek file, and that some paired tests sum three difficulty variants whose problem identifiers overlap; both issues are absent from the experiment reported here, which uses one problem set per comparison and a single fixed scorer.

G PROMPT AND STATE-REPORT SENSITIVITY

The readout wording is selected on separate calibration bases using a fixed tie-break order and then frozen. This is important because wording is not a harmless presentation detail: v8 records model-dependent reversals between direct, recent, and current queries, while v9.1 shows that round-trip verification can fail even when continuation remains possible. We therefore treat a report as a probe outcome, not as a gold-standard observation of the hidden computation.

A future stronger assay would verify the state through a continuation that requires the state but does not ask the model to name it, and would keep this verification separate from the primary estimand. Such an assay could test whether the v9.1 report/use gap persists without conditioning the interruption comparison on a downstream natural-language readout.

H REGISTEROPS FACTORIAL IN FULL

Table 4 gives every cell of the within-example operator factorial that Figure 2 plots as a heatmap.

Table 4: RegisterOps 6×6 operator factorial, continuation accuracy (%) within base example. Rows are the intervening Task-B operator and columns the resumed Task-A operator; the diagonal is the exact-match condition.

$O_B \backslash O_A$	+1	+2	+3	-1	-2	-3
+1	62.5	45.0	45.0	100.0	100.0	87.5
+2	87.5	65.0	47.5	100.0	100.0	97.5
+3	97.5	62.5	55.0	100.0	87.5	92.5
-1	57.5	47.5	47.5	100.0	97.5	77.5
-2	40.0	47.5	40.0	97.5	100.0	72.5
-3	67.5	47.5	42.5	85.0	37.5	85.0

Paired contrasts: exact match $+6.08\,\mathrm{pp}$ $[1.67, 10.58]$; magnitude match $+4.17\,\mathrm{pp}$ $[1.25, 7.08]$; sign match $+0.97\,\mathrm{pp}$ (interval includes zero).

I REGISTEROPS IN FULL: JOINT OUTCOMES, CONTRASTS, AND REPLICATION

Table 5 gives the full 2×2 joint distribution of the two probes on the interrupted RegisterOps histories, from which the two off-diagonal rates in the main text are read. The two dissociation cells are populated at similar magnitude, so the failure is not concentrated in one direction.

Table 5: Joint outcome of the state report (R) and the continuation (U) on the same interrupted histories, Qwen2.5-7B, $n = 1440$ paired cells. The two off-diagonal rows are the report–continuation dissociations.

R	U	n	Share
1	1	845	58.68%
1	0	259	17.99%
0	1	204	14.17%
0	0	132	9.17%

Table 6 gives every contrast behind Figure 2(b), including the stratification by the sign of the resumed operator that produces the near-zero aggregate sign-match estimate.

Table 6: Operator-relation contrasts, matched minus unmatched continuation accuracy (pp), within base example. Intervals are cluster bootstrap over the 40 base histories; n counts factorial cells.

Relation	Stratum	n	Δ (pp)	95% CI
exact	all	1440	$+6.08$	$[1.67, 10.58]$
exact	resumed $+$	720	$+6.00$	$[-1.33, 13.33]$
exact	resumed $-$	720	$+6.17$	$[1.83, 10.50]$
magnitude	all	1440	$+4.17$	$[1.25, 7.08]$
magnitude	resumed $+$	720	-1.25	$[-5.42, 2.71]$
magnitude	resumed $-$	720	$+9.58$	$[5.42, 13.96]$
sign	all	1440	$+0.97$	$[-1.67, 3.89]$
sign	resumed $+$	720	$+14.44$	$[8.89, 19.72]$
sign	resumed $-$	720	-12.50	$[-16.39, -8.89]$

Table 7 gives the same factorial on the two replication models. Only the continuation branch is usable on these runs: their clean state probe was never recorded and their interrupted probe never returns the paused value, so the report–continuation dissociation cannot be measured on them.

Table 7: Cross-model operator contrasts. Continuation accuracy (%) within base example on the same 40-base 6×6 design. The last column records whether the state probe is usable on that run.

Model	Clean	Interrupted	Exact (pp)	Magnitude	Sign	Probe
Qwen2.5-7B (bf16)	100.0	72.85	$+6.08$	$+4.17$	$+0.97$	yes
Qwen2.5-3B (fp16)	97.5	89.93	-10.42	$+1.04$	-4.58	no
Llama-3.1-8B (fp16)	100.0	95.88	$+4.95$	$+5.00$	$+0.53$	no

J WORKED EXAMPLE: FIXED REPORT, CHANGED CONTINUATION

The following case is copied from the recorded v7 paired row `v7base-0000`, not reconstructed from the aggregate statistics. It is included to make the central counterfactual concrete.

The row is informative for three reasons. The report branch returns the paused value exactly, so the error is not a failure to name S_A. The clean branch establishes that the same model and same continuation operation can produce the correct result without the intervening task. The interrupted branch then returns 5, which is consistent with applying the recent Task-B operator to the paused A state rather than applying the resumed A operator. This is an illustrative instance of the aggregate error taxonomy; it is not, by itself, a causal estimate.

The full `resume_messages` for this row contain the controlled system instruction, the Task-A update and pause, the Task-B update with an ‘OK‘ response, and the ‘RESUME Task A‘ cue. The final continuation request is not changed between the clean and interrupted branches. The paired analysis in the main paper aggregates this same type of comparison over 40 bases and all 6×6 operator pairs.

Table 8: Core RegisterOps behavior. Report and continuation branches share the same resumed history. The off-diagonal rows are the report–continuation dissociations: $\mathrm{R} = 1, \mathrm{U} = 0$ shows a correct report is not sufficient for use, $\mathrm{R} = 0, \mathrm{U} = 1$ that it is not necessary.

Metric	Clean	Interrupted
State report accuracy	100.00%	76.67%
Continuation accuracy	100.00%	72.85%
$\Pr(\mathrm{R} = 1, \mathrm{U} = 0)$	—	17.99%
$\Pr(\mathrm{R} = 0, \mathrm{U} = 1)$	—	14.17%

Table 9: A recorded RegisterOps example. The suspended state and resumed operation are fixed; the interrupted continuation differs from the clean continuation.

Field	Recorded value
Task-A initialization	$A = 2$, then pre-operation $+1$
Suspended state	$S_A = 3$
Intervening task	Task B starts at $B = 7$, applies $+1$, verifies successfully
Resumed operation	$O_A = +1$
Gold continuation	$3 + 1 = 4$
Clean report / continuation	`A=3` / `A=4` (both correct)
Interrupted report / continuation	`A=3` (correct) / `A=5` (incorrect)
Protocol flags	A protocol, B execution, B verification, and resume all passed

K WORKED CLOSURE-CONTROL EXAMPLE

A second recorded case comes from v6.1, base `v6-000000`, replicate 0, with $S_A = 3$, $O_A = +1$, and gold continuation 4. In the open, explicit-B, matched-filler, and random-number branches, the recorded continuation is `A=4`. In the `closure_text` branch, the continuation is also `A=4`, but the direct state report becomes `A=4` rather than the gold paused state `A=3`. Thus the added closure text changes the report behavior in this example without changing the correct continuation.

This case does not prove that closure is irrelevant. It shows why the paper separates the state report from the continuation and why the closure claim is based on paired rates rather than one favorable example. The full v6.1 report compares 720 paired examples: closure text improves continuation over open by 4.58 percentage points, but performs identically to matched filler on continuation, while explicit B improves continuation by 8.89 points over open. The paired results therefore support a broader recency/externalization interpretation rather than a unique semantic-closure mechanism.

L CLOSURE AND CUE CONTROLS IN FULL

Table 11 reports every v6.1 condition on the same 720 bases, and Table 12 gives the paired contrasts that the main text summarizes. Two properties are worth noting. First, closure text raises continuation relative to the open prompt but is exactly tied with a matched filler turn ($0.00\,\mathrm{pp}$, CI $[-1.53, 1.67]$), so the gain is attributable to adding a turn rather than to its semantic content. Second, among the cue variants, `old_restore` produces a 14.4% readout rollback to Task A's initial value while `precise_recent` produces none, which is the reported example of instruction-conditioned state reporting.

M CROSS-MODEL CONGRUENCY: LIMITS

Three limits apply. First, `side_ok` is condition-dependent (≈ 0.76 for conflict and congruent conditions but ≈ 0.55 for the unrelated condition in Qwen2.5-7B), so the conditions are not matched on intervening-task success; the main-text conclusion therefore rests on the congruent-versus-conflict comparison, where the two rates are close, rather than on the unrelated condition. Second, the `strong_resume` condition changes the resume prompt as well as the intervening state, so its large negative effect is a prompt-protocol difference rather than a mechanism result. Third, the direction of the conflict effect is model-dependent, so the experiment delimits the phenomenon rather than establishing a universal interference penalty.

Table 10: GSM8K re-entry: exact-match final-answer accuracy (%) on the same 1000 problems per model. Δ_{math} is the paired difference from the clean continuation and p its exact McNemar value; the probe columns are the state probe with and without the interruption.

Model	Clean	Unrel.	Math	Probe clean	Probe intr.	Δ_{math} (p)
Qwen2.5-14B	94.9	95.7	95.8	64.1	31.1	+0.90 (2.2e-01)
Qwen2.5-7B	95.1	94.3	91.7	89.6	81.3	−3.40 (1.2e-04)
Qwen2-7B	90.0	87.8	83.9	88.6	78.7	−6.10 (5.7e-08)
Llama-3.1-8B	91.5	90.7	88.3	93.2	91.5	−3.20 (4.8e-03)
Qwen3-4B	93.6	93.1	90.4	78.6	70.4	−3.20 (7.7e-05)
Qwen2.5-3B	89.6	89.1	84.1	79.9	74.1	−5.50 (4.2e-06)

Table 11: v6.1 condition summary, 720 paired bases each. Readout is the direct state probe and continuation is the resumed operation; the last column is the off-diagonal $\Pr(\mathrm{R}=1,\mathrm{U}=0)$.

Condition	Readout	Continuation	$\Pr(\mathrm{R}=1,\mathrm{U}=0)$
none (clean)	100.00	100.00	—
open	91.67	78.75	17.08
matched filler	99.31	83.33	16.39
closure text	50.00	83.33	4.31
explicit B	75.83	87.64	6.39
random number	61.25	86.94	3.89
cue: precise recent	54.72	73.89	10.28
cue: old restore	85.56	77.08	18.06
cue: closure precise	86.94	75.97	17.64

N SECOND PROTOCOL: BRANCH-VERIFIED CONFLICT

An independent branch-verified protocol ($n = 300$ bases per condition, seven conditions) confirms both the conflict null result and the report–continuation dissociation with all gates passing. For continuation, `same_congruent` is 94.67% and `same_conflict` is 95.00%, a difference of 0.33 pp, while the direct-capture rate is at most 1.0% in every condition. The `strong_resume` condition again dissociates sharply: continuation stays at 92.00% while the readout collapses to 29.67%.

The prefix-level patch in this run does not support a localization claim. Three of its four controls (clean, congruent, and reverse) transplant a state that is identical to the target state, so they are no-ops by construction and only validate determinism. Only the random-donor control performs a genuine replacement, with donor–target gaps of 1 to 7 register units, and its mean effect is at most 0.003 in probability against a ceiling baseline of 0.985 for the target readout; individual trials move by up to 0.23. A mean null measured at ceiling is weak evidence, so we report this protocol for its behavioural dissociation and not as a test of state transplantation.

O ATTENTION ANALYSIS IN FULL

Table 14 reports the per-layer share of the answer position's attention that lands on the span carrying the suspended Task-A value, for all three replay models. The table lists each model's four largest reductions together with the number of layers that move in each direction. The reduction is broad rather than a single-layer artefact, and it is present in every model even though the peak layer differs.

The head-level picture matches the layer-level one. Most cells move in the same direction, but the movement is not uniform: 74.4% of Qwen2.5-7B's 784 cells fall, so roughly a quarter rise. That is why the main text describes the interruption as reallocating attention rather than switching it off, and why the layer-averaged share is the quantity used for the reported contrasts.

Table 12: v6.1 paired effects on the same 720 bases. Positive continuation deltas favour the second condition.

Contrast	Metric	Δ (pp)	95% CI
closure text − open	continuation	$+4.58$	$[2.78, 6.39]$
closure text − matched filler	continuation	0.00	$[-1.53, 1.67]$
explicit B − open	continuation	$+8.89$	$[6.39, 11.39]$
explicit B − random number	continuation	$+0.69$	$[-1.39, 2.78]$
closure text − open	readout	-41.67	$[-45.28, -38.19]$

Table 13: Cross-model numerical state conflict, continuation accuracy (%) with $n = 2000$ bases per condition. `side_ok` is the fraction of conflict-condition trials whose intervening task was verified, and `proto` the fraction passing the full protocol. Rows four and five are independent data seeds for the same model; the last row is a protocol boundary.

Model	None	Congr.	Confl.	Strong	Unrel.	Δconfl.	side_ok	proto
Qwen3-4B	21.50	11.45	11.50	10.70	13.50	-10.00	0.78	1.00
Llama-3.1-8B	11.95	9.75	8.75	7.75	18.90	-3.20	0.78	1.00
Qwen2.5-7B	35.15	35.40	35.05	23.75	33.30	-0.10	0.77	1.00
Qwen2.5-7B seed 2027	33.35	—	33.45	22.15	—	$+0.10$	0.77	1.00
Qwen2.5-7B seed 2028	33.05	—	33.70	22.45	—	$+0.65$	0.75	1.00
Qwen2.5-0.5B[†]	5.90	3.45	3.45	3.05	3.30	-2.45	0.05	0.92

[†]Protocol boundary, not evidence: the intervening task is verified on only 5.4% of conflict trials, so this model cannot be pooled with the others.

P A RECORDED GSM8K RE-ENTRY FAILURE

Table 16 reproduces one recorded GSM8K trial in full, so that the natural-task failure is visible as a completion rather than as a rate. The reference solution is shown up to its second step; the model then either continues it directly or continues it after a second math problem has been inserted.

Both branches resume from the same prefix with the same instruction, and only the interrupted one contains an intervening math problem. The clean branch treats \$45.00 as the total for the pants; the interrupted branch treats it as the price of one pair and doubles it. This is the natural-task counterpart of the operator-structured error taxonomy in Section 6.2: the suspended quantity survives as a token sequence but is re-interpreted on re-entry.

Q CONDITION-LEVEL STATE REPORT AND CONTINUATION

Figure 5 reports the four v9.1 priming conditions in full. The top row gives direct report accuracy and continuation accuracy, and the bottom row gives the two off-diagonal joint rates on the same conditions. The sections of the main text that discuss the symbolic experiment draw on these profiles.

R SYMBOLIC PRIMING CONTROLS IN FULL

PRIMING-MATCHED CONTROLS DELIMIT SYMBOLIC GENERALIZATION

The priming-matched ShuffleOps experiment does not support a universal positive symbolic interruption effect: the common-prefix contrast is $+0.93\,\mathrm{pp}$ for 7B, $-5.32\,\mathrm{pp}$ for 14B, and $-1.16\,\mathrm{pp}$ for 32B. For Qwen2.5-7B a generic recent swap reproduces the gain of the intervening Task-B segment entirely ($+19.4\,\mathrm{pp}$ over an unprimed clean baseline, with a Task-B-specific remainder of $+0.5\,\mathrm{pp}$ and CI $[-1.5, 2.7]$), and the remainder is non-zero only in Qwen2.5-32B ($-1.9\,\mathrm{pp}$). Under verification-conditioned complete cases the 7B comparison is not estimable and the 14B contrast changes to $+0.83\,\mathrm{pp}$, so a probe intended to validate state recovery can change the question being answered. Averaged over cross-task conditions, 7B reports the state on 66.8% of cells but continues correctly on 91.0%, 14B on 71.3% and 70.6%, and 32B on 100% and 78.7%; calibration coverage is unequal, so these are diagnostic rather than a model-size curve. The safe statement is not that interruptions never affect symbolic computation, but that the earlier positive signal was not identified

Table 14: Attention re-entry by layer. Share of the answer position's attention on the span holding the suspended value, for the four layers with the largest reduction in each model. The two middle columns give the total number of layers that fall and rise.

Model	Layers down / up	Layer	Clean	Interrupted	Δ
Qwen2.5-7B	20 / 8 of 28	20	0.0996	0.0710	−0.0286
		8	0.0442	0.0230	−0.0212
		17	0.0690	0.0496	−0.0194
		15	0.0660	0.0484	−0.0175
Qwen2.5-3B	31 / 5 of 36	32	0.2975	0.2464	−0.0511
		21	0.1480	0.1068	−0.0412
		25	0.2107	0.1721	−0.0386
		33	0.0877	0.0525	−0.0352
Llama-3.1-8B	30 / 2 of 32	18	0.0863	0.0653	−0.0210
		16	0.1176	0.0977	−0.0200
		30	0.0733	0.0605	−0.0128
		15	0.0476	0.0351	−0.0124

Table 15: Attention re-entry by head. The per-head paired change is averaged over the 40 replayed base histories. "Cells down" is the share of layer–head cells whose mean paired change is negative, a descriptive counterpart to the interval-based count reported in the main text.

Model	Layers $\times$ heads	Cells	Cells down	Mean $\lvert\Delta\rvert$
Qwen2.5-7B	28×28	784	74.4%	0.0179
Qwen2.5-3B	36×16	576	77.8%	0.0198
Llama-3.1-8B	32×32	1024	82.1%	0.0096

as an interruption effect by the available control. Figure 6 and Appendix R give the full priming diagnostic and condition profiles.

Figure 6 reports the priming diagnostic and the priming-matched switch contrast that the main text summarises.

S SHUFFLEOPS: FILTERING SENSITIVITY

The v9.1 verification probe is downstream of the assigned priming condition and itself measures a report-like behavior. Conditioning on verification success changes sample composition by condition. In Qwen2.5-7B, no complete four-condition pairs survive the verification filter, while all 864 paired cells per condition remain under the common-prefix analysis. In Qwen2.5-14B, the main effect is $+0.83$ pp under complete cases and -5.32 pp under the common-prefix analysis, a sign reversal. We therefore treat the common-prefix result as the primary descriptive comparison and the complete-case result as sensitivity analysis, while explicitly noting that the preregistered positive gate fails.

T SHUFFLEOPS GATE AND FILTERING LEDGER

Table 18 records the calibration and gate quantities that determine how the v9.1 results are interpreted. A low calibration score is not repaired by choosing a more favorable wording after the main run. The selected wording is frozen per run, and its valid count is reported. The 'base_ok' gate remains false for all three models because the preregistered continuation/readout requirements are not jointly met.

The common-prefix analysis uses 864 cells per condition for 7B and 432 cells per condition for 14B and 32B. The complete-case paired counts are 0, 120, and 210, respectively. These denominators explain why the 7B conditional estimate is not available and why the 14B estimate is especially sensitive to filtering.

Table 16: A recorded GSM8K re-entry failure (`gsm8k_train_4047`, Qwen2.5-7B). The suspended state is the accumulated pants total, \$45.00 for two pairs. The clean branch keeps it as a subtotal; the math-interrupted branch re-reads it as a per-pair price and doubles it.

	Content
Reference (shown)	The first pair of pants is \$30.00 and the second pair is \$15.00 so the pants cost $30 + 15 =$ \$45.00
Resume cue	Continue the solution from where it stopped and give the final answer.
Clean branch	… Step 5: Add the costs of the skirts, pants, and blouses. $60 + 45 + 75 =$ \$180.00. `#### 180`
Math-interrupted branch	… Pants: 2 pairs of pants $\times$ \$45.00 = \$90.00 … $60 + 90 + 75 =$ \$225.00. `#### 225`
Gold	\$180.00

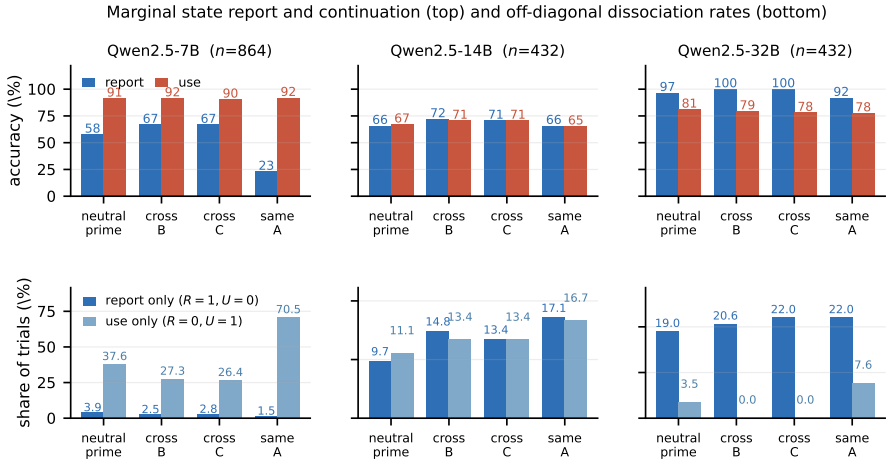


Figure 5: Condition-level state report and continuation under the common-prefix analysis (top), with the two off-diagonal dissociation rates $\Pr(\mathrm{R}=1,\mathrm{U}=0)$ and $\Pr(\mathrm{R}=0,\mathrm{U}=1)$ (bottom). The 7B profile is dominated by use-only trials and the 32B cross-task profile by report-only trials, so the cross-model reversal is a change in which behavioral branch fails rather than a shift in two marginal averages.

U Counterfactual patch experiment in full

The patch assay was the first attempt to locate a residual state mechanically. For Qwen2.5-7B with $N = 216$, a state-donor patch at the answer frontier swaps the answer almost completely: the collision-answer probability rises by 0.972 while the target-answer probability falls by 0.962. At the three earlier anchors the same substitution moves neither answer by more than 0.001, and the donor and target residual vectors are nearly collinear there, so those zeros reflect a near-no-op intervention rather than a localization result.

What the experiment does establish is that the late effect tracks answer identity. The state donor and operator donor have different latent decompositions but the same alternative final answer, and at the final layer larger Qwen2.5 models show approximately complete transfer for both donor types, including models with no behavioral interruption failure on the v8 task. The late causal effect is therefore neither specific to the interruption failure nor sufficient evidence for a state–operator binding representation, and patching claims require controls that separate the named latent factor from answer-proximal content (Makelov et al., 2024; Zhang and Nanda, 2024).

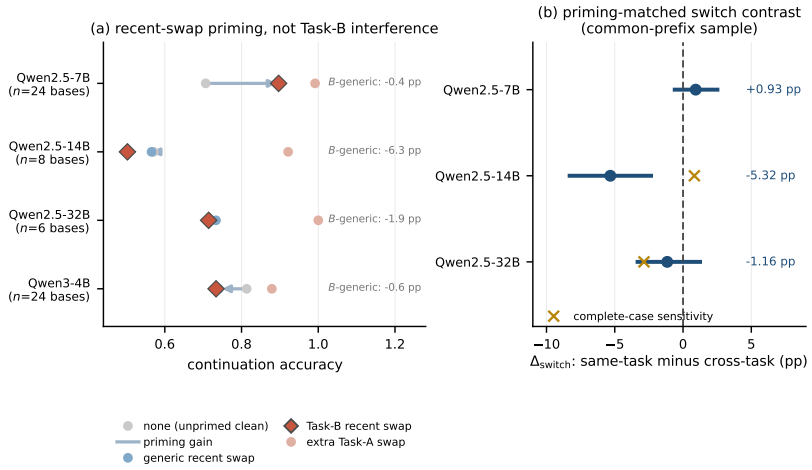


Figure 6: Symbolic re-entry. (a) A recent swap before the probe raises continuation over the unprimed clean condition, but a generic recent swap on a third object set reproduces nearly all of that gain, so the v9 interruption effect is recency priming rather than Task-B interference. (b) After priming is matched, the same-task-minus-cross-task contrast is near zero or negative; crosses mark verification-conditioned complete-case estimates.

Table 17: Priming-matched ShuffleOps interruption contrast. Common-prefix analysis preserves randomized/paired condition assignment; complete-case analysis conditions on a downstream verification probe.

Model	Common-prefix Δ	95% CI	Complete-case Δ
Qwen2.5-7B	$+0.93$ pp	$[-0.75, 2.66]$	not estimable
Qwen2.5-14B	-5.32 pp	$[-8.45, -2.20]$	$+0.83$ pp
Qwen2.5-32B	-1.16 pp	$[-3.47, 1.39]$	-2.86 pp

V PATCH EFFECTS BY ANCHOR AND STEP CALIBRATION

Table 19 gives the per-anchor patch numbers behind Figure 7: the change in collision-answer probability, the change in target-answer probability, and the donor–target cosine that determines whether the substitution is a genuine intervention at all. Table 20 gives the step calibration that bounds how far the six-operation task can be read as a capability probe.

At one step the task is close to ceiling and the conflict condition is at ceiling, so a conflict effect cannot be detected; by three and five steps the task is near floor for both conditions. This is why the main text treats the six-operation conflict contrast as a probe-sensitive comparison rather than a capability measurement.

W EARLIER PATCH CONTROLS AND STEP CALIBRATION

An earlier factor-isolation round used same/random/reverse/self-patch controls at all 28 layers of Qwen2.5-7B, with baseline and patched generations sharing one decoding path. Here the self, same-question, and reverse controls transplant a state identical to the target and therefore act as determinism checks; self-patching reproduces the baseline text in $2800/2800$ layer-target pairs. The random-donor control is the genuine replacement, and it changes correctness from a $38/100$ baseline to 36–$38/100$ at every layer, with a same-donor advantage between -1.0 and $+1.0$ pp and Holm-corrected exact tests at $p = 1.0$ throughout. With a 38% baseline this null has headroom, but the sample is 100 targets, the intervals are pointwise rather than simultaneous, and a null does not establish equivalence.

The same round also calibrated task length. Clean continuation is 96.5% at one operation, 34.5% at three, and 8.0% at five, so the three-operation setting sits near a capability boundary. The three settings used different data seeds, so the drop cannot be attributed to length alone, and we use it only as a caveat on the absolute level of the congruency results.

Table 18: v9.1 calibration and gate quantities. The same-A columns are computed on verification-passing rows, not on the common-prefix sample, so they answer a conditional gate question and are not comparable to the common-prefix profiles in Figure 5.

Model	Bases	Calibration valid/attempted	Same-A use†	Same-A report†	Base gate
Qwen2.5-7B	24	1/108	0.861	0.083	false
Qwen2.5-14B	12	14/108	0.707	0.620	false
Qwen2.5-32B	12	23/108	0.779	0.888	false

†Verification-passing denominators differ sharply across conditions: for 7B the same-A gate cells are 36 rows over 6 of 24 bases (report 0.083) and the all-condition gate divides 1008 valid rows by 3456 recorded rows, whereas the common-prefix profiles in Figure 5 use 864 cells per condition over all 24 bases (same-A report 0.229, same-A use 0.919). Both quantities are reported because the gate is defined on verified rows while the descriptive comparison is defined on the assigned conditions.

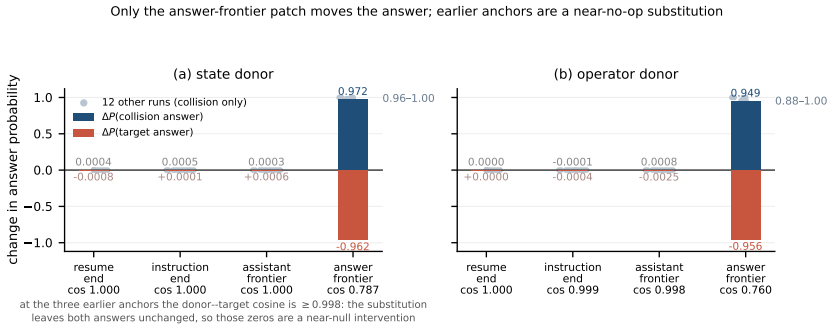


Figure 7: Counterfactual patch effects by temporal anchor. Bars are the primary Qwen2.5-7B run ($N = 216$) at each anchor's discovery-selected layer; grey dots are the 12 other runs (collision answer only) and the grey bar spans their range. At the answer frontier the substitution swaps the answer almost completely; at the three earlier anchors the donor–target cosine is ≥ 0.998 and neither answer probability moves, so those zeros reflect a near-no-op intervention rather than a localization result.

X Mechanistic Interpretation Boundaries

The v8 result should not be read as proof that intermediate reasoning states do not exist, and it also does not establish that no transplantable state exists before the answer frontier. It bounds one claim: the only large effect of the tested single-position intervention appears at the answer frontier, where same-answer collision donors with different latent decompositions transfer equally well, so that effect tracks answer identity rather than a state–operator binding.

Three design details bound the reading of Figure 7. First, each anchor is patched at its own layer selected on the discovery split, so the cross-anchor comparison varies temporal position and layer together rather than position alone; for the primary 7B run the selected layers are $12, 12, 12, 27$ for the state donor and $0, 4, 15, 27$ for the operator donor. Second, the earlier anchors are not a fair test of localization. Taking, for each run and anchor, the layer at which donor and target are least similar, the donor–target cosine across all 14 recorded runs is at least 0.994 at the resume boundary, 0.926 at the instruction boundary, and 0.942 at the assistant frontier, against 0.75–0.84 at the answer frontier; at those earlier anchors the substitution leaves both the target and the collision answer probability unchanged. The early-anchor nulls are therefore nulls of the intervention rather than localization results. Third, the confirmation sample for the primary run is $N = 216$ triples while the 12 comparison runs use $N = 72$ or 216 and differ in dtype and sampling seed; the answer-frontier effect holds across all of them. Distributed representations, multi-position interventions, attention patterns, and KV-cache state therefore remain open, and a fair early-anchor test would need donors whose activation distance from the target is matched across anchors.

Y Hypothesis Falsification Ledger

Table 19: Counterfactual patch effects by temporal anchor, primary Qwen2.5-7B run ($N = 216$). A near-unit cosine means the donor activation is almost identical to the target, so the intervention is close to a no-op and a null there is not evidence against localization.

Donor	Anchor	Layer	ΔP(collision)	ΔP(target)	Cosine
state	`resume_end`	12	$+0.0004$	-0.0008	0.9997
state	`instruction_end`	12	$+0.0005$	$+0.0001$	0.9996
state	`assistant_frontier`	12	$+0.0003$	$+0.0006$	0.9998
state	`answer_frontier`	27	$+0.9719$	-0.9618	0.7868
operator	`resume_end`	0	0.0000	0.0000	1.0000
operator	`instruction_end`	4	-0.0001	-0.0004	0.9991

Table 20: Step calibration on the six-operation register task. Accuracy is the continuation rate in each condition; the side-task column is the fraction of intervening tasks that verified, which bounds how far the conflict contrast can be read.

Steps	Condition	n	Accuracy	Side task OK
1	none	200	0.965	1.00
1	same-conflict	200	1.000	0.84
3	none	200	0.345	1.00
3	same-conflict	200	0.355	0.79
5	none	200	0.080	1.00
5	same-conflict	200	0.055	0.735

Hypothesis	Stronger control	Outcome
Task B overwrites Task A's numeric state	Congruent vs. conflicting B states; capture analysis	Not supported; conflict-specific harm is absent or small.
Semantic closure restores Task A	Closure cue vs. matched filler and unrelated numeric turn	Not supported; semantic closure has no unique benefit.
Resume-boundary residual stores transplantable suspended state	Counterfactual patch at resume boundary	Not testable under this assay: donor and target activations are nearly collinear at the resume boundary, so the patch is a near-no-op.
Late patch identifies state/operator binding	Same-answer factor-isolation donors across temporal anchors	Falsified under this assay; effect appears at answer frontier and collapses to answer identity.
Symbolic task switch has a universal positive cost	Round-trip priming-matched same-A vs. cross-B/C	Not supported; 7B/32B near zero, 14B negative.