Escaping Answer Anchoring: Fixing Shortcut Tokens in Reasoning Models via Context-Replay Gaps

Anonymous authors
Paper under double-blind review

Abstract

Long-form reasoning models are increasingly deployed as autonomous agents, but longer inference and larger context windows do not ensure they use the right evidence. We find that even after extensive reinforcement learning with verifiable rewards (RLVR), reasoning models remain anchored to their earlier answers: when a previous attempt is wrong, they typically restate it rather than produce a different answer. Outcome-level RLVR supplies no direct token-local label, while standard supervised fine-tuning (SFT) weights valid target positions uniformly; neither identifies which tokens depend on the preceding solution. We introduce the context-replay gap, a per-token diagnostic that compares each token's log-likelihood when the preceding solution is present in context with its log-likelihood when it is absent. Tokens whose likelihood is strongly increased by the replayed solution are separated from those receiving little or negative likelihood change. The gap guides Context-Replay Guided Tuning (CRT) by selecting tokens for an ordinary negative log-likelihood (NLL) loss. This targeted supervision improves mathematical reasoning across model families and benchmarks. Behavioral interventions show that CRT reduces dependence on previously stated answers while preserving sensitivity to the reasoning that supports them. The context-replay gap therefore provides both a diagnostic of answer anchoring and a practical signal for post-training.

1 Introduction

Reasoning models increasingly solve difficult problems through long autoregressive trajectories, motivating their use as components of more general-purpose agents (Wei et al., 2022; Shao et al., 2024; DeepSeek-AI, 2025). Yet longer inference and larger context windows do not guarantee that a model uses the right evidence. Models can over-rely on nearby reasoning tokens (Chen et al., 2024), be distracted by irrelevant premises (Shi et al., 2023), underuse relevant information in the middle of a long context (Liu et al., 2024), or remain anchored to an earlier answer (Huang et al., 2024; Zhang et al., 2025b). These phenomena are not necessarily instances of one internal mechanism. They nevertheless expose a shared behavioral bottleneck: autoregressive prediction can favor convenient context over task-relevant evidence, even when the latter is available.

Post-training can improve reasoning without necessarily removing this bottleneck. The base models in our main experiments are themselves RLVR-trained reasoning checkpoints, and the shortcut behavior is still clearly present: at the first multi-pass transition, when the previous answer is wrong, these models still follow it 86% (1.5B) and 96% (7B) of the time; Figure 1 shows a two-pass example. How RLVR produces its gains remains disputed: large-sample studies suggest it partly reallocates probability among behaviors already available in the base model (Chen et al., 2026b), while other work finds frontier expansion under longer training (Liu et al., 2025). Either way, the outcome-level signal does not identify which target tokens depend on the preceding solution. Standard SFT is equally uninformative about this dependence because it applies NLL uniformly across target positions.

We investigate this distinction with a context-replay contrast. Given a question x and an offline response y, we compute each target token's log-likelihood under a regular context x and a replay

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.

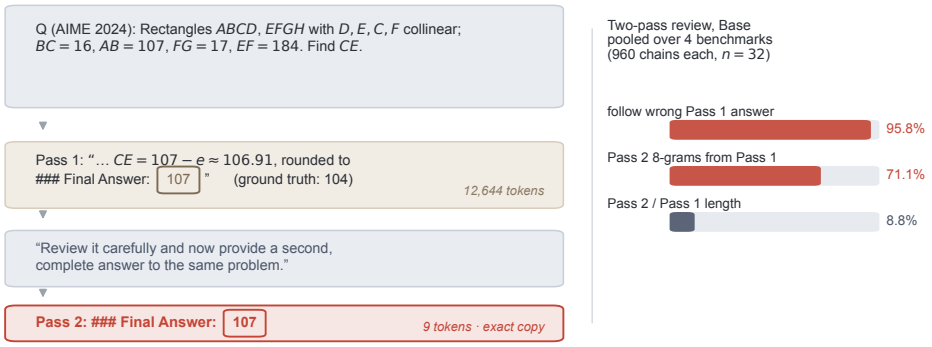


Figure 1: **Does a second turn help the model revise a wrong answer?** Base RLVR (JustRL-1.5B) answers an AIME 2024 geometry problem 107 (ground truth 104) after 12,644 tokens; asked to review and re-answer, it emits nine tokens restating 107. Right: pooled two-pass review over four benchmarks.

context (x, y, x). The difference between these log-likelihoods measures how much the preceding assistant turn helps predict that token. Unlike random masking, this paired contrast preserves the target response and turns sensitivity to prior generated content into a per-token diagnostic. We call this quantity the *context-replay gap*. The gap is the core of *Context-Replay Guided Tuning* (CRT): self-generated reasoning is replayed in context to expose per-token gaps, which select target tokens for a regular-context NLL objective. On a training set of only 1,000 queries, this recipe yields substantial gains in fewer than two epochs (Table 2). An update-40 matched control sharpens the result: with the identical 10% fraction and configuration, selecting the highest-gap tokens instead reverses accuracy by about 11 pp on every benchmark, so the *direction* of the measured gap—not sparsity alone—carries the effect.

The gap also raises a mechanistic question. A history-conditioned target may be easier because the model reuses intermediate computation, but it may simply copy the prior answer. We therefore intervene on turn boundaries, history correctness, answer spans, reasoning order, and individual reasoning steps, using CRT. The tuned model does not become selectively resistant to wrong history: a wrong turn remains strongly harmful under every protocol. What changes is the answer channel. The benefit of a correct history and the main effect of answer correctness both contract by roughly two thirds to half, whereas we detect no change in the main effect of reasoning correctness. Following the previous answer falls, and accuracy after a wrong source in a full second turn rises from 7.89% to 30.26%.

The same pattern appears inside one turn, with no history at all. A fixed tentative answer is written into the model's own assistant turn before any reasoning, and the model continues generation from this prefix (Figure 2): Base is strongly controlled by the pre-stated answer and the tuned model far less, while reasoning still drives both models.

Our main contributions are:

- **A token-level diagnosis of answer-anchoring dependence.** The context-replay gap scores the same target token with and without a replayed prior solution, isolating how much that solution changes token likelihood while holding the target response fixed.
- **A behavioral test of answer anchoring.** Cross-turn and same-turn interventions show that CRT makes isolated prior answers substantially less controlling.
- **Single-turn gains from targeting answer-anchoring tokens.** Across three model families and four benchmarks, optimizing the tokens identified by the context-replay gap improves ordinary single-turn reasoning in regular context. Detailed ablations vary token-ranking direction, scoring policy, training context, replay content, and rule-based revision-marker selection to isolate the role of token choice.

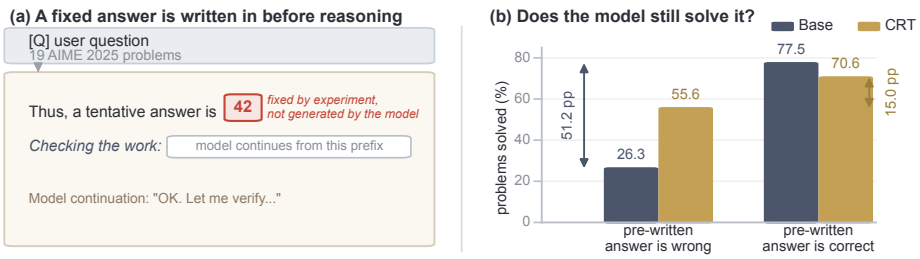


Figure 2: **Does an answer written first anchor a model within one turn?** Before any reasoning, a fixed tentative answer is written directly into the model's own assistant turn, followed by "Checking the work:"; the model continues generation from this prefix. Bars: answer-only fraction solved, grouped by whether the pre-written answer is wrong or correct. Each bracket shows how much success rises when that answer is correct rather than wrong: 51.2 pp for Base but only 15.0 pp for CRT, so the pre-stated answer controls Base far more strongly.

2 Related Work

Post-training objectives and token selection. RLVR and off-policy policy distillation assign supervision at the response level (Chen et al., 2026b; Liu et al., 2025; Chen et al., 2026a); standard SFT instead optimizes every target token. Masked Thought randomly removes parts of the conditioning trace to reduce dependence on nearby reasoning (Chen et al., 2024). CRT uses a different unit of selection: it holds the target response fixed, measures each token's sensitivity to a replayed prior solution, and uses that contrast to choose tokens for ordinary regular-context SFT.

Context dependence in reasoning. Irrelevant premises, context position, and decomposition all change reasoning behavior (Shi et al., 2023; Yang et al., 2025; Liu et al., 2024; Yan et al., 2025; Xu et al., 2026). These studies establish that context matters; CRT asks how the model's own completed solution changes each token's likelihood in a fixed target response. The regular-versus-replay contrast turns that dependence into a training selector rather than an attention score or an inference-time context perturbation.

Self-correction and answer anchoring. Prompted self-correction often fails without external feedback (Huang et al., 2024; Stechly et al., 2025), and revision can inherit prompt bias or answer inertia (Zhang et al., 2025b; Yang & Jia, 2026). Prior context can also induce token reuse or misinformation following (Niu et al., 2025; Gao et al., 2026), while attention, attribution, and faithfulness studies ask how generated reasoning relates to model computation (Wu et al., 2025; Zhang et al., 2025a; Du et al., 2026; Arcuschin et al., 2025; Boppana et al., 2026). CRT instead improves ordinary single-turn reasoning without requiring an external review prompt at inference time. Its cross-turn and same-turn interventions further show weaker answer anchoring.

3 Method

3.1 Context-replay token gaps

Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ be fixed question–response pairs. For target token $y_{i,t}$, we denote its log-likelihoods in the regular and replay contexts by ℓ^{reg} and ℓ^{replay}, respectively:

$$\ell_{i,t}^{\text{reg}} = \log \pi(y_{i,t} \mid x_i, y_{i,<t}), \tag{1}$$

$$\ell_{i,t}^{\text{replay}} = \log \pi(y_{i,t} \mid x_i, y_i, x_i, y_{i,<t}), \tag{2}$$

$$g_{i,t} = \ell_{i,t}^{\text{replay}} - \ell_{i,t}^{\text{reg}}. \tag{3}$$

The replay context inserts the complete assistant turn y_i before reissuing the question; the target response is identical in both contexts. A positive gap means that the prior assistant turn increases the target token's likelihood.

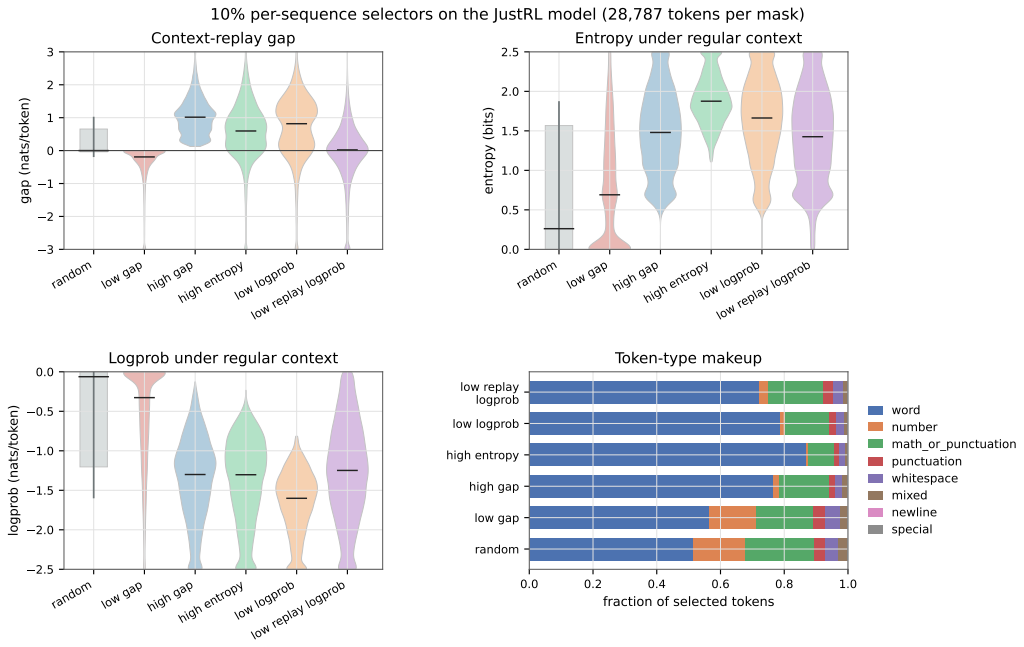


Figure 3: **Score distributions and token composition for 10% selectors.** The gray Random reference uses the full token pool as the population analogue of uniform sampling: boxes span the 10th–90th percentiles, whiskers the 5th–95th percentiles, and bars the median. Colored violins show the selected tokens. The final panel compares token-type fractions; High entropy uses the regular context.

3.2 Low-gap token selection

Our recommended recipe uses the gap to select the target tokens for supervision. A low gap means that replaying the previous response provides little likelihood gain for the token; a negative gap means that the replayed response suppresses it relative to the regular context. Independently within each response, let $M_i^{(f)}$ contain the $\lceil fT_i \rceil$ tokens with the smallest measured gap, breaking ties by token position. We optimize ordinary regular-context NLL:

$$\mathcal{L}_{\text{low-gap}} = -\frac{\sum_{i,t} M_{i,t}^{(f)} \log \pi_\theta(y_{i,t} \mid x_i, y_{i,<t})}{\sum_{i,t} M_{i,t}^{(f)}}. \tag{4}$$

We use $f = 0.1$ unless otherwise stated. Appendix Table 10 reports the complete $f = 0.005$–1.0 sweep at updates 40 and 50 together with generated-response lengths.

3.3 What the selectors capture

The gap would be less informative if it merely recovered tokens that are already improbable, uncertain, or easy to identify from lexical cues. We therefore compare equal-budget 10% masks along three axes: their scores under regular and replay contexts, their token-type composition, and their overlap with explicit revision language. Appendix Table 7 gives each selector's operational definition.

Figure 3 tests whether low-gap selection merely recovers tokens that are already difficult in regular context. It does not. Low-gap tokens cluster at negative gaps, meaning that replay suppresses them relative to regular context, yet their regular-context likelihoods are substantially higher than those selected by low log probability. The other selectors do not isolate the same replay effect: high entropy spans a broad range of gaps, while low replay log probability includes both positive and negative gaps. Token composition provides a second distinction. High entropy and low log probability mostly select words, whereas low gap retains more numbers, mathematical symbols, and punctuation, producing a

composition closer to random selection. Thus low gap captures replay sensitivity rather than simply low probability or high uncertainty.

We next ask whether it mainly detects explicit reconsideration language. On 287,739 tokens from 32 JustRL training-pool responses, we identify 5,125 **revision markers** (e.g., "Wait," "But," and "Hmm"). Table 1 measures how strongly each selector concentrates these markers.

Table 1: **Revision-marker composition of 10% token selectors.** The selectors are applied to 287,739 tokens from 32 JustRL training-pool responses. Capture is the fraction of all revision markers selected; share is the marker fraction within the selected mask; enrichment divides that share by the marker fraction in the full pool. Random is the mean over 1,000 stratified draws.

Selector	Marker capture (%)	Marker share (%)	Enrichment
Random 10%	10.0	1.8	1.0×
Low gap	40.9	7.3	4.1×
High entropy	40.3	7.2	4.0×
Low replay log prob.	**50.4**	**9.0**	**5.0**×
Low log prob.	33.7	6.0	3.4×

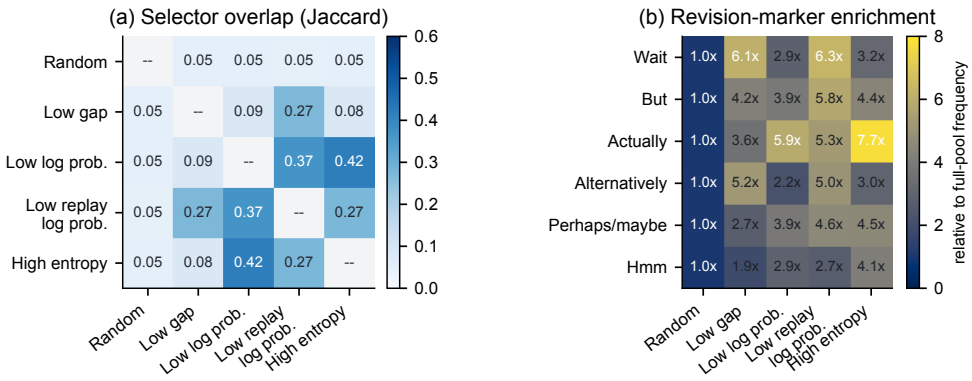


Figure 4: **Selector overlap and revision-marker composition.** (a) Jaccard overlap, $|A \cap B|/|A \cup B|$, among equal-budget masks; Random uses one fixed draw. (b) Marker-family enrichment over full-pool frequency; Random averages 1,000 draws. High entropy uses the regular context.

Revision language is enriched, but it does not define the low-gap mask. Low gap captures 40.9% of markers at a 10% token budget, compared with 10.0% under stratified random selection; high entropy captures a similar 40.3%, while low replay log probability captures 50.4%. Yet low gap overlaps only weakly with low log probability and high entropy (J=0.09/0.08) and moderately with low replay log probability (J=0.27; Figure 4a). The marker families also differ: low gap most strongly enriches *Wait* and *Alternatively*, whereas high entropy favors *Actually*. Together, the score, content, and overlap analyses show that replay sensitivity selects a distinct token subset that includes, but is not reducible to, visible revision cues.

4 Experimental Setup

Model and data. We study three RLVR-trained reasoning checkpoints: 1.5B JustRL-Nemotron (JustRL), 7B AceReason-Nemotron (ACE), and Qwen3-4B-OpenMath (QW). We compare six NLL selectors with two off-policy reward-optimization baselines: Group Relative Policy Optimization (GRPO) and cross-entropy (CE) reward optimization (β=0.01). For a controlled comparison, all objectives optimize the same fixed set of Base-model rollouts. CE is an unpaired-DPO-style binary objective on reference-relative sequence log-likelihood, using verifier correctness as the label. We include CE because prior work finds it more sample-efficient than GRPO (Chen et al., 2026a). We randomly sample 1,000 queries from DeepScaleR (Luo et al., 2025). For each query, the corresponding Base model generates 16 rollouts, yielding 16,000 training trajectories per model family. NLL

optimization uses a global batch of 512 rollouts and typically reaches its best validation performance around update 50, or about 1.6 epochs over the rollout set.

NLL baselines. Full-token SFT applies regular-context NLL to all response tokens. The 10% rows select per response: Random SFT uses fixed pseudorandom positions; High entropy uses the largest frozen-reference regular-context entropy; Low regular or replay log probability uses the smallest target likelihood in that context; and Low gap uses the smallest g. All optimize regular-context NLL.

Evaluation. We evaluate four datasets—AIME 2024, AIME 2025, HMMT February, and HMMT November—with 30 problems each and 32 sampled responses per problem ($n = 32$). Accuracy is the mean binary correctness over the sampled pool. Table 2 gives the sample accuracy for each family; Appendix A's *Training and decoding settings* paragraph gives the full settings. Where checkpoint sweeps are available, we select on AIME 2025, leaving the other three benchmarks outside checkpoint selection.

5 Results

5.1 Main results

Table 2 compares the same reward and NLL objectives across JustRL, ACE, and QW.

Table 2: **Main results on four mathematical reasoning benchmarks.** Pooled sample accuracy (%) over 30 questions per benchmark. Rows are grouped by optimization objective (*Reward*: GRPO and a cross-entropy reward, β=0.01; *NLL*: regular-context NLL, token selection ratio in parentheses), and **Avg** is the macro over benchmarks. Decoding uses t=0.7 for JustRL and t=0.6 for ACE/QW. Checkpoint selection uses AIME 2025 where a sweep is available; the other columns are not used for selection. Rows come from separate runs, so this is a descriptive sweep rather than a paired significance test.

Objective	Method	AIME 2024	AIME 2025	HMMT	HMMT Nov	Avg
(a) JustRL-Nemotron-1.5B						
	Base	68.9	59.6	37.9	42.7	52.3
Reward	GRPO	69.5	61.2	39.2	41.3	52.8
	CE (β=0.01)	70.0	61.4	38.3	42.8	53.1
NLL	Full-token SFT (100%)	69.6	61.3	37.7	41.5	52.5
	Random SFT (10%)	68.2	59.6	39.0	42.0	52.2
	High entropy (10%)	69.3	60.0	37.6	41.3	52.0
	Low log prob. (10%)	69.3	61.5	37.5	42.2	52.6
	Low replay log prob. (10%)	70.6	64.0	42.4	44.5	55.4
	Low gap (10%)	**74.1**	**66.2**	**43.3**	**47.7**	**57.8**
(b) ACE: AceReason-Nemotron-1.1-7B						
	Base	72.4	65.7	43.4	57.4	59.7
Reward	GRPO	72.6	64.9	44.9	57.1	59.9
	CE (β=0.01)	73.2	65.4	44.5	57.2	60.1
NLL	Full-token SFT (100%)	73.8	62.6	42.2	57.4	59.0
	Random SFT (10%)	74.3	65.0	43.5	57.5	60.1
	High entropy (10%)	73.9	67.0	44.6	55.6	60.3
	Low log prob. (10%)	73.8	64.2	43.3	56.0	59.3
	Low replay log prob. (10%)	74.3	69.1	45.6	57.2	61.5
	Low gap (10%)	**74.6**	**71.9**	**49.2**	**59.6**	**63.8**
(c) QW: Qwen3-4B-OpenMath						
	Base	84.9	81.1	56.0	66.6	72.2
Reward	GRPO	86.8	81.2	57.8	65.7	72.9
	CE (β=0.01)	87.3	83.6	67.6	70.4	77.2
NLL	Full-token SFT (100%)	86.7	83.3	57.3	64.4	72.9
	Random SFT (10%)	86.0	79.0	57.1	65.4	71.9
	High entropy (10%)	86.5	82.3	55.4	64.4	72.1
	Low log prob. (10%)	86.3	79.4	54.4	66.3	71.6
	Low replay log prob. (10%)	87.5	83.3	60.0	70.4	75.3
	Low gap (10%)	**89.0**	**86.3**	**60.2**	**70.4**	**76.5**

Across all three model families, low gap has the highest observed four-benchmark average among NLL selectors. Its four-benchmark average exceeds Base by 5.5 pp on JustRL, 4.1 pp on ACE, and

4.3 pp on QW. It also has the highest observed score in every JustRL and ACE benchmark cell; on QW, it leads the NLL rows and both AIME columns. Low replay log probability is the closest NLL alternative in all three families. This main-table proximity is consistent with Figure 4: low replay log probability has the largest mask overlap with low gap (J=0.27) and a broadly similar revision-marker enrichment profile. By contrast, two baselines that might also appear promising—selecting high-entropy tokens or tokens with low regular-context log probability—remain near or below Base across the three families. The gain therefore does not follow from generic hard-token mining.

Reward optimization and low-gap NLL exhibit different transfer profiles even though they use the same rollout data. This is clearest on QW: CE gains 11.6 pp over Base on HMMT, whereas low-gap NLL gains 4.2 pp. The ordering reverses on AIME 2024 and 2025, where low-gap NLL gains $4.1/5.2$ pp compared with CE's $2.4/2.5$ pp. Low-gap NLL never observes verifier rewards, so these contrasting benchmark gains suggest that reward optimization and selective NLL may emphasize different transferable knowledge in the same trajectories.

We examine the selector and training choices more closely through the detailed ablations in Section 7.

6 Analysis of Reasoning Behavior

The two questions posed by Figures 1 and 2 separate whether another turn actually produces correction from whether a previously written answer controls subsequent generation. The evidence supports four findings.

6.1 Analysis 1: CRT follows incorrect prior answers less often

Figure 1 asks whether a second turn helps revise a wrong answer. Although CRT already starts above Base at Pass 1, repeated previous-only replay provides further gains through Pass 5 (Figure 5a): macro accuracy increases by 1.85 points for JustRL and 1.28 points for ACE; on ACE, the latter exceeds Base's 0.73-point gain. These multi-pass gains remain smaller than CRT's single-pass margins over Base (6.26 points for JustRL and 4.09 for ACE). Figure 5c suggests one contributor: on ACE, CRT raises recovery, especially from Pass 1 to Pass 2, but also raises degradation at later transitions, which may limit compounding across passes.

CRT also follows a wrong source answer less often and lowers source 8-gram overlap, especially for JustRL (Figure 5b). Here *answer following* means that the next answer matches the immediately preceding answer; *8-gram reuse* measures source-reasoning overlap. We define *recovery* as $P(\text{next correct} \mid \text{source wrong})$ and *degradation* as $P(\text{next wrong} \mid \text{source correct})$.

Table 3: **Macro-average per-pass accuracy (%) across four benchmarks.** Pass 1 answers the question without history; each later pass receives only the immediately preceding response before the repeated question. P1→P5 is the absolute change in percentage points. Per-benchmark results and settings are in Appendix Table 6 and its preceding settings paragraph.

Family	Model	P1	P2	P3	P4	P5	P1→P5
JustRL	Base	51.56	52.01	53.20	53.52	54.04	+2.48
	CRT	**57.83**	**57.23**	**58.61**	**59.05**	**59.68**	+1.85
ACE	Base	59.95	60.47	60.68	60.81	60.68	+0.73
	CRT	**64.04**	**65.00**	**65.00**	**65.21**	**65.31**	+1.28

Across the four model groups, aggregate accuracy changes by only 0.7–2.5 pp from Pass 1 to Pass 5 (Table 3); the per-benchmark results are in Table 6. Thus most of the gain comes from training rather than repeated review.

Within a single regular-context response, we next quantify explicit answer changes. On JustRL, boxed-only extraction raises the pooled rate of any later correct answer from 1.01% to 1.58%, while template matching raises it from 2.60% to 3.29% (Table 4), with the largest absolute increases on AIME25 and HMMT. Because even explicit answer declarations can refer to intermediate subproblems, we treat both views as descriptive sensitivity analyses rather than population self-correction.

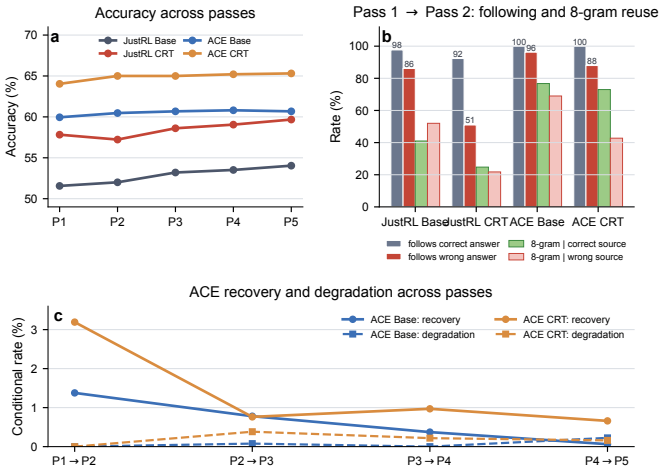


Figure 5: **Five-pass behavior across JustRL and ACE.** (a) Absolute macro accuracy at each pass. (b) Answer following and reasoning 8-gram reuse from Pass 1 to Pass 2, split by whether the Pass 1 source answer is correct. (c) ACE recovery and degradation at each transition, macro-averaged over the four benchmarks.

Table 4: **Single-turn answer transitions within JustRL responses (%).** Among responses whose first detected answer is wrong, we report whether any later answer is correct and whether the final detected answer is correct. *Any later* counts a response if at least one subsequent detected answer is correct, regardless of how the response ends; *Final* counts it only when the last detected answer is correct. *Boxed only* uses `\boxed{...}` occurrences; *template matching* additionally accepts textual declarations explicitly marked `final` or `correct`.

		Boxed only		Template matching	
Dataset	Model	Any later	Final	Any later	Final
AIME24	Base	1.49	1.00	4.03	3.55
	CRT	1.59	1.59	4.63	4.63
AIME25	Base	0.90	0.64	4.71	4.22
	CRT	1.56	1.10	5.57	4.97
HMMT	Base	1.25	0.75	1.91	1.41
	CRT	2.56	1.74	3.45	2.45
HMMT-Nov	Base	0.55	0.46	1.00	0.82
	CRT	0.50	0.40	0.90	0.80
All	Base	**1.01**	**0.68**	**2.60**	**2.20**
	CRT	**1.58**	**1.18**	**3.29**	**2.80**

The broader output distribution changes more: CRT raises JustRL accuracy by 5.6 pp, reduces the normalized extracted-answer distribution's entropy from 1.70 to 1.43 bits, increases its modal mass by 4.7 pp, and lengthens responses by 7.8K tokens (11.5K among wrong responses). Thus its outputs are longer and more concentrated without a comparable increase in within-response answer revision.

6.2 Analysis 2: Cross-turn answer anchoring weakens

The cross-turn interventions separate the prior answer from its reasoning. Relative to (Q, Q), correct history changes Base/CRT accuracy by $+38.82/+19.08$ pp, while wrong history changes it by $-40.79/-38.16$ pp. Correspondingly, the crossed-factorial answer-correctness effect contracts from 34.54 pp to 9.87 pp, while the reasoning-correctness effect is nearly unchanged (42.43 pp versus 41.45 pp). Equal-length neutral history and answer-only controls have little effect on CRT, but wrong reasoning remains harmful when answer literals are masked and when its steps are shuffled (Table 8). Surface reuse decreases (Figure 5b); forced scoring is Base-only and the attention shift correlational

(Tables 15 and 16). Thus CRT weakens cross-turn answer anchoring while retaining order-sensitive reasoning influence; after a wrong source, a fully regenerated second turn improves accuracy from 7.89% to 30.26%. Full conditions are in Appendix Tables 11 and 12.

6.3 Analysis 3: The same selective shift appears within one turn

With a wrong tentative answer alone, CRT solves 55.6% versus Base 26.3%, but complete wrong reasoning floors both (CRT/Base: 0.5%/0.3%). Correct reasoning helps both, and fixing only the last wrong step has similar effects (14.80 pp/13.49 pp; Appendix Table 14). Thus CRT resists an isolated wrong answer, not a complete wrong reasoning trace.

6.4 Analysis 4: Length does not explain the gains

Longer outputs often accompany gains but are insufficient: marker-only training adds 55.6% tokens for only +1.77 pp, while excluding those markers retains +5.21 pp with 36.7% growth (Table 9). The sweep exposes a trade-off: update-40 $f = 0.8$ gives +4.53 pp for +21.8% tokens, whereas update-50 $f = 0.1$ gives +6.62 pp (Table 10). Thus length alone does not account for the observed gains; token selection matters.

7 Ablations and Diagnostics

Ablation axes. Table 5 varies CRT's gap scoring/ranking, NLL context, replay content, and revision-marker inclusion. Appendix Table 7 defines selector and training-path labels; the linked *Ablation masks* paragraph defines replay-content and marker-mask variants, including the complete expanded-v2 lexicon.

Table 5: **JustRL AIME 2025 ablations.** Sampled accuracy; Δ is relative to current-policy low-gap SFT.

Axis	Variant	Acc. / Δ (pp)
Default	**Regular-context SFT, current-policy low gap**	**66.2** / –
Frozen score/rank	Reference-policy low gap	65.6 / −0.6
	Reference-policy near-zero gap	62.1 / −4.1
	Reference-policy high gap	51.1 / −15.1
	Reference-policy high absolute gap	62.0 / −4.2
Training context	Replay-context SFT, current-policy low gap	64.2 / −2.0
	Replay-context SFT, reference-policy low gap	64.1 / −2.1
Online direction	Regular-context SFT, high gap	53.5 / −12.7
Replay content	Answer segment only	64.4 / −1.8
	Boxed answer only	60.6 / −5.6
	Final-answer sentence only	61.5 / −4.7
Marker content	All revision markers	49.2 / −17.0
	Low gap ∩ revision markers	61.4 / −4.8
	Low gap \ revision markers	64.8 / −1.4
	Low gap \ revision markers (expanded v2)	63.5 / −2.7
	Low-gap half within revision markers	59.9 / −6.3
	High-gap half within revision markers	40.7 / −25.5

Online beats frozen scoring (66.2% vs. 65.6%), and regular-context NLL beats replay-context SFT (64.2%). Answer-segment replay retains 64.4% versus 60.6–61.5% for answer-only variants. Marker-only training yields 49.2%; removing markers retains 63.5–64.8%, and their low-gap half beats the high-gap half (59.9% vs. 40.7%).

8 Conclusion

CRT uses context-replay gaps to select training tokens and improves accuracy across all three model families. Behavioral interventions associate these gains with weaker answer anchoring while preserving sensitivity to ordered reasoning.

References

Iván Arcuschin, Jett Janiak, Robert Krzyzanowski, Senthooran Rajamanoharan, Neel Nanda, and Arthur Conmy. Chain-of-thought reasoning in the wild is not always faithful. *arXiv preprint arXiv:2503.08679*, 2025.

Siddharth Boppana, Annabel Ma, Max Loeffler, Raphael Sarfati, Eric Bigelow, Atticus Geiger, Owen Lewis, and Jack Merullo. Reasoning theater: Disentangling model beliefs from chain-of-thought. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://icml.cc/virtual/2026/poster/66648.

Changyu Chen, Xiting Wang, Ting-En Lin, Ang Lv, Yuchuan Wu, Xin Gao, Ji-Rong Wen, Rui Yan, and Yongbin Li. Masked thought: Simply masking partial reasoning steps can improve mathematical reasoning learning of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5872–5900, 2024. doi: 10.18653/v1/2024.acl-long.320. URL https://aclanthology.org/2024.acl-long.320/.

Changyu Chen, Xiting Wang, and Rui Yan. Extreme region policy distillation, 2026a. URL https://arxiv.org/abs/2605.25582.

Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *Advances in Neural Information Processing Systems*, 38:57654–57689, 2026b.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Kevin Du, Alexander Hoyle, Laura Ruis, and Acyr Locatelli. Legibility is not interpretability: Comparing judged and actual importance in chain-of-thought reasoning. In *Conference on Language Modeling*, 2026. URL https://arxiv.org/abs/2609.04194.

Muhan Gao, Zih-Ching Chen, and Kuan-Hao Huang. The first drop of ink: Nonlinear impact of distracting information in long-context reasoning. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://icml.cc/virtual/2026/poster/65095.

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*, 2024. URL https://arxiv.org/abs/2310.01798.

Mingjie Liu, Shizhe Diao, Ximing Lu, Jian Hu, Xin Dong, Yejin Choi, Jan Kautz, and Yi Dong. ProRL: Prolonged reinforcement learning expands reasoning boundaries in large language models. *arXiv preprint arXiv:2505.24864*, 2025.

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. doi: 10.1162/tacl_a_00638.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y. Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Li Erran Li, Raluca Ada Popa, and Ion Stoica. DeepScaleR: Surpassing O1-preview with a 1.5b model by scaling RL. Notion Blog, project page, 2025.

Jingcheng Niu, Xingdi Yuan, Tong Wang, Hamidreza Saghir, and Amir H. Abdi. Llama see, llama do: A mechanistic perspective on contextual entrainment and distraction in llms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pp. 16218–16239, 2025. doi: 10.18653/v1/2025.acl-long.791.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 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 *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 31210–31227, 2023. URL https://proceedings.mlr.press/v202/shi23a.html.

Kaya Stechly, Karthik Valmeekam, and Subbarao Kambhampati. On the self-verification limitations of large language models on reasoning and planning tasks. In *International Conference on Learning Representations*, 2025. URL https://iclr.cc/virtual/2025/poster/31014.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. Retrieval head mechanistically explains long-context factuality. In *International Conference on Learning Representations*, 2025. URL https://iclr.cc/virtual/2025/poster/30373.

Mayi Xu, Ke Sun, Jianhao Chen, Qiankun Pi, Guixin Su, Yunfeng Ning, Yongqi Li, Yuanyuan Zhu, Ming Zhong, Jiawei Jiang, and Tieyun Qian. Recursive short-to-long generalization for multi-hop reasoning. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2118–2128, 2026. doi: 10.1145/3805712.3809697.

Shaotian Yan, Chen Shen, Wenxiao Wang, Liang Xie, Junjie Liu, and Jieping Ye. Don't take things out of context: Attention intervention for enhancing chain-of-thought reasoning in large language models. In *International Conference on Learning Representations*, 2025. URL https://iclr.cc/virtual/2025/poster/29384.

Minglai Yang, Ethan Huang, Liang Zhang, Mihai Surdeanu, William Yang Wang, and Liangming Pan. How is llm reasoning distracted by irrelevant context? an analysis using a controlled benchmark. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 13329–13347, 2025. doi: 10.18653/v1/2025.emnlp-main.674.

Yuqing Yang and Robin Jia. When do LLMs admit their mistakes? understanding the role of model belief in retraction. In *Conference on Language Modeling*, 2026. URL https://arxiv.org/abs/2505.16170.

Jue Zhang, Qingwei Lin, Saravan Rajmohan, and Dongmei Zhang. From reasoning to answer: Empirical, attention-based and mechanistic insights into distilled deepseek r1 models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 3985–4002, 2025a. doi: 10.18653/v1/2025.emnlp-main.198.

Qingjie Zhang, Di Wang, Haoting Qian, Yiming Li, Tianwei Zhang, Minlie Huang, Ke Xu, Hewu Li, Liu Yan, and Han Qiu. Understanding the dark side of llms' intrinsic self-correction. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pp. 27066–27101, 2025b. doi: 10.18653/v1/2025.acl-long.1314.

AI use statement

In this work, we used generative AI tools to refine hypotheses, draw figures, support qualitative data analysis, and interpret experimental results. We did not use generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for or assist with mathematical proofs, or translate the manuscript; the remaining required-disclosure tasks are not applicable. The authors reviewed all AI-assisted work, tested AI-assisted implementations, and verified every reported value against the underlying evaluation artifacts. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

A Experimental Details

Training and decoding settings. Launchers use a batch size of 512, PPO mini-batch size 512, micro-batch size one per GPU, maximum prompt/response lengths of 17,408/15,360 tokens, and eight GPUs. Selector runs use learning rate 10^{-6}, with checkpoints saved every 5 or 10 steps through steps 40–60.

JustRL uses temperature 0.7, top-p=0.9, and a maximum model length of 65,536. ACE uses temperature 0.6, top-p=0.95, and maximum length 65,536. QW uses temperature 0.6, top-p=0.95, top-k=20, and maximum length 81,920.

Table 6: **Per-benchmark accuracy (%) for all five-pass runs.** P1→P5 is the absolute change in percentage points.

Family	Model	Benchmark	P1	P2	P3	P4	P5	P1→P5
JustRL	Base	AIME24	68.12	69.58	71.35	71.77	72.40	+4.28
		AIME25	58.54	58.96	59.69	59.79	60.21	+1.67
		HMMT	36.88	37.08	38.33	38.44	39.27	+2.39
		HMMT-Nov	42.71	42.40	43.44	44.06	44.27	+1.56
		Avg	**51.56**	**52.01**	**53.20**	**53.52**	**54.04**	**+2.48**
	CRT	AIME24	74.11	74.79	76.15	75.99	76.20	+2.09
		AIME25	66.20	63.75	65.10	65.68	66.20	0.00
		HMMT	43.28	42.66	44.48	44.64	45.99	+2.71
		HMMT-Nov	47.71	47.71	48.70	49.90	50.31	+2.60
		Avg	**57.83**	**57.23**	**58.61**	**59.05**	**59.68**	**+1.85**
ACE	Base	AIME24	72.29	72.50	72.71	72.71	72.50	+0.21
		AIME25	66.67	67.50	67.92	68.33	67.92	+1.25
		HMMT	43.44	43.75	43.85	43.96	43.96	+0.52
		HMMT-Nov	57.40	58.13	58.23	58.23	58.33	+0.93
		Avg	**59.95**	**60.47**	**60.68**	**60.81**	**60.68**	**+0.73**
	CRT	AIME24	74.58	76.15	76.25	76.35	76.35	+1.77
		AIME25	70.94	71.98	72.08	72.40	72.60	+1.66
		HMMT	49.17	49.48	49.38	49.79	49.90	+0.73
		HMMT-Nov	61.46	62.40	62.29	62.29	62.40	+0.94
		Avg	**64.04**	**65.00**	**65.00**	**65.21**	**65.31**	**+1.28**

Table 7: **Operational definitions of experimental labels.** Selection fractions are applied independently within each response unless stated otherwise.

Label	Operational definition
Training objectives and main-table selectors	
Base	The original RLVR-trained checkpoint, evaluated without additional fine-tuning.
Full-token SFT (100%)	Regular-context NLL on every valid response token.
Random SFT (10%)	Regular-context NLL on a fixed pseudorandom 10% of valid token positions in each response; the mask is independent of likelihood, entropy, and gap.
High entropy (10%)	The 10% of tokens with highest next-token predictive entropy under the frozen reference policy in regular context.
Low log prob. (10%)	The 10% of target tokens with lowest log-likelihood ℓ^{reg} in regular question–answer context.
Low replay log prob. (10%)	The 10% of target tokens with lowest log-likelihood ℓ^{replay} when the full prior response and repeated question are inserted before the same target response.
Low gap (10%)	The 10% of tokens with smallest $g = \ell^{\text{replay}} - \ell^{\text{reg}}$; this is the default CRT selector.
GRPO	Off-policy Group Relative Policy Optimization using verifier rewards over groups from the same fixed Base-model rollout set used by the other objectives.
CE ($\beta = 0.01$)	An off-policy, unpaired-DPO-style binary objective on reference-relative sequence log-likelihood; verifier-correct and verifier-incorrect trajectories provide positive and negative labels, and β scales the classification logit.
Gap and training-path ablations	
Current-policy gap	Recomputed online with the evolving policy during training.
Reference-policy gap	Computed with the frozen reference model and therefore fixed during training.
High gap	Selects the largest values of g; it reverses the low-gap ranking at the same token budget.
Near-zero gap	Selects the smallest values of $\lvert g \rvert$.
High absolute gap	Selects the largest values of $\lvert g \rvert$, irrespective of sign.
Regular-context SFT	Computes the selected-token NLL under the ordinary question–answer input $(x, y_{<t})$.
Replay-context SFT	Uses the same selected tokens but computes their NLL under the replay input $(x, y, x, y_{<t})$.
Answer segment only	When scoring the gap, replaces the full replayed response with the text after `</think>`.
Boxed answer only	Replays only the extracted `\boxed{...}` expression.
Final-answer sentence only	Replays the standardized sentence "The final answer is `\boxed{...}`."
Revision markers	Rule-matched reconsideration spans. The *Ablation masks* paragraph lists the complete base lexicon (including *Wait*, *Actually*, *But*, and *Hmm*) and the additional checking, explicit-error, and alternative-approach phrases in expanded v2.

Continued on next page

Table 7 continued

Label	Operational definition
Behavioral protocols and metrics	
Five-pass, previous-only	Pass 1 answers the question without history. Each later pass sees only the immediately preceding response followed by the repeated question.
Answer following	The next response's extracted answer is grader-equivalent to the source response's answer.
8-gram reuse	Fraction of next-response reasoning 8-grams that also occur in the source reasoning.
Recovery / degradation	$P(\text{next correct} \mid \text{source wrong})$ / $P(\text{next wrong} \mid \text{source correct})$.
Boxed only / template matching	Boxed only detects `\boxed{...}` occurrences. Template matching also detects answer declarations explicitly marked *final* or *correct*.
Any later / Final	Given a wrong first detected answer, Any later asks whether any subsequent detected answer is correct; Final asks whether the last detected answer is correct.
Answer entropy / modal mass	Shannon entropy of the normalized extracted-answer distribution / share assigned to its most frequent answer.
Controlled context interventions	
No / full history	No history supplies only the current question. Full history inserts the complete prior response before repeating the question.
Reasoning-only / answer-only	Reasoning-only retains the prior derivation while masking every answer-equivalent span; answer-only supplies only a standardized sentence containing the prior final answer.
Mismatch	Holds the prior reasoning fixed while replacing only its terminal answer with an answer of opposite correctness.
Dummy / answer-masked / shuffled	Matched controls respectively use neutral equal-length text, replace answer literals in place, or permute the answer-masked reasoning steps.
Same-turn intervention	Writes a fixed tentative answer and cue before reasoning, then lets the model continue the same assistant turn without a new user question.

Ablation masks. The replay-content variants change only the history used to score the selector. *Answer segment only* removes the delimited reasoning block and preserves the remaining answer text; *boxed answer only* retains only `\boxed{...}`; *final-answer sentence only* retains "The final answer is `\boxed{...}`." For token-content controls, *all revision markers* supervises every matched marker, while the intersection and difference rows respectively keep or remove markers from the bottom-10% low-gap mask. The base lexicon covers *Wait*, *Actually*, *However*, *reconsider*, *recheck*, *I made a mistake*, *That was incorrect*, *But*, *Alternatively*, *perhaps*, *maybe*, and *Hmm*. Expanded v2 additionally covers *Let me check/verify/reconsider/re-evaluate*, *Let's check/reconsider*, *Hold on*, *Wait a minute*, *On second thought*, *I realize/was wrong/made an error*, *This was incorrect*, *This/That cannot be right*, *Check again*, *Double-check*, and *Another/A different approach*. The final two rows rank only revision-marker tokens by gap and retain the lower or upper half. These masks do not select equal numbers of tokens.

B Matched Context Budget and Response Length

Table 8 reports the main matched comparison. All rows share the same 19 questions and eight matched seeds, and every history-bearing row uses the same Q–A–Q frame. Dummy is a neutral repeated phrase truncated to the reasoning length; answer-masked replaces answer literals in place; shuffled permutes the masked reasoning; answer-only and intact supply just the reported answer or the complete original turn. Character counts are matched rather than tokenizer-token counts, and the generated-token budget is unconstrained.

Table 8: **Matched context-budget controls after a wrong source.** Base or CRT receives a repeated-question input whose inserted A_1 is edited as listed, then generates A_2. Cells report A_2 final-answer accuracy over the same 19 questions and eight matched rollouts; history-bearing variants are matched by character count.

Condition	Base (%)	CRT (%)
No history	53.29	61.84
Dummy text, equal characters	28.95	59.21
Wrong answer only	49.34	59.21
Reasoning present, answer masked	8.55	25.66
Wrong intact history	7.24	33.55
Steps shuffled	7.89	17.11

Observed pattern. Equal-length dummy text sharply reduces Base accuracy but leaves CRT close to its no-history baseline. A wrong answer by itself also has little effect on CRT. In contrast, every condition containing the wrong reasoning causes a large drop, even after answer literals are masked. Shuffling those steps is especially harmful to CRT, showing that the model remains sensitive to the order of the supplied reasoning.

Boundary cues and full regeneration. Opening the reasoning boundary after a wrong source with a neutral continuation changes 2.01% of Base answers and 5.30% of CRT answers; *Wait* or *Reflect* continuations leave recovery at 0.00% for Base and 1.32–2.63% for CRT, and closed reflection yields nothing. A full unconstrained second turn behaves differently: accuracy after a wrong source is 7.89% for Base but 30.26% for CRT, answer changes are 13.38% versus 46.67%, source following falls from 82.89% to 54.61%; after a correct source CRT accuracy is 81.58% versus Base 91.45%. The five-pass recovery values are smaller because that protocol uses previous-only continuation rather than full regeneration.

Degeneration checks. Aggressive high-gap optimization can produce repeated markers such as *Hmm* at later checkpoints, where means are dominated by outliers; we inspect medians, the 95th percentile (P95), the count of distinct boxed answers, and per-thousand-token marker density.

Revision-subset length. The low-gap–marker intersection and set differences under the base and expanded marker lexicons are evaluated on AIME 2025; Table 9 reports the main accuracy–length comparison. Mean characters follow the same ordering; generated-token counts use the checkpoint tokenizer with special tokens excluded.

Table 9: **Accuracy versus response length on AIME 2025.** Each JustRL checkpoint generates regular question-only responses. Columns report sampled final-answer accuracy and mean generated tokens; the rows compare supervision on revision-marker intersections and set differences against Base and the undivided low-gap selector. The three revision-subset variants use step-40 pools with n=32 per question; Base and the undivided step-40 low-gap reference use n=64.

Training subset	Accuracy (%)	Mean tokens
Base	59.58	14,321
Low gap ∩ revision markers	61.35	**22,284**
Low gap \ revision markers	64.79	19,577
Low gap \ revision markers (expanded v2)	63.54	19,250
Undivided low gap (step 40)	**65.68**	21,730

Fraction sensitivity and response length. Table 10 gives the complete AIME 2025 sweep. Along the update-40 $n = 64$ sequence ($f = 0.05, 0.1, 0.6, 0.8, 0.9$), mean length falls from 23,021 to 16,218 tokens while accuracy declines overall from 65.94% to 62.40%; the $+0.62$ pp change from $f = 0.6$ to 0.8 is a small local reversal. Selecting all tokens further approaches Base length and accuracy, and update 50 shows the same endpoint decline. Thus larger fractions shorten responses and gradually reduce accuracy, with local sampling variation.

Table 10: **Complete JustRL low-gap fraction sweep on AIME 2025.** Each cell reports regular-context sampled accuracy and generated-response length. All pools contain 30 questions and use the official grader. For update 40, $f = 0.05/0.1/0.6/0.8/0.9$ use 64 rollouts per question; for update 50, $f = 0.05/0.1/0.2$ use 64; the remaining pools use 32. Both updates use t=0.7 and top-p=0.9.

	Update 40		Update 50	
f	Mean tok.	Acc.	Mean tok.	Acc.
0.005	23,349	63.85	25,496	63.54
0.01	23,728	**65.94**	26,058	65.42
0.02	23,582	64.69	26,106	63.85
0.05	23,021	**65.94**	24,682	65.36
0.1	21,730	65.68	23,241	**66.20**
0.2	20,861	64.17	21,626	65.42
0.4	20,432	64.17	20,941	63.33
0.6	19,499	63.49	20,575	63.33
0.8	17,440	64.11	17,490	62.92
0.9	16,218	62.40	16,006	62.50
1.0	14,594	61.25	14,109	60.21
Base	14,321	59.58	14,321	59.58

C Comparison Across Intervention Protocols

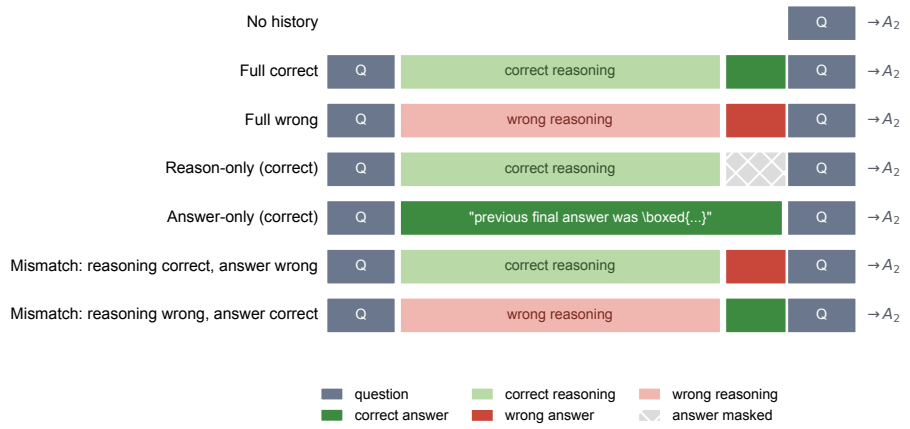


Figure 6: **Context composition of the cross-turn conditions.** Each row shows the input before Base or CRT generates A_2. A history-bearing row has the form (Q, A_1, Q), where A_1 contains the indicated Base-produced reasoning and/or terminal answer. In reasoning-only conditions, every answer-equivalent span is replaced in place by `[ANSWER_REMOVED]`. Answer-only supplies the sentence "The previous final answer was `\boxed{·}`." Mismatch conditions keep the reasoning source fixed and replace only the terminal answer with one of opposite correctness.

Table 11: **Boundary-condition accuracy and contrasts.** Base or CRT generates A_2 from the listed input. Accuracy is measured over the same 19 questions and eight matched rollouts per question; Δ is relative to the duplicated-question input (Q, Q).

Input	Base acc.	Base Δ	CRT acc.	CRT Δ
Question only, Q	56.58	+3.95	66.45	+0.66
Duplicated question, (Q, Q)	52.63	–	65.79	–
Full wrong history, (Q, A_1, Q)	11.84	−40.79	27.63	−38.16
Full correct history, (Q, A_1, Q)	91.45	+38.82	84.87	+19.08

Table 12: **Cross-turn input intervention and generated-output accuracy.** For each of 19 questions, we construct A_1 from Base-produced correct or wrong reasoning (with answer-equivalent spans masked) and append either a correct or wrong terminal answer. Base or CRT receives (Q, A_1, Q) and generates a new response A_2; cells report A_2 final-answer accuracy over eight matched rollouts per question. The answer main effect is the average gain from changing only the terminal answer from wrong to correct; the reasoning main effect analogously changes only the reasoning source.

Constructed A_1 history	Base A_2 acc.	CRT A_2 acc.
Correct reasoning + correct answer	88.82	82.89
Correct reasoning + wrong answer	59.21	70.39
Wrong reasoning + correct answer	51.32	38.82
Wrong reasoning + wrong answer	11.84	31.58
Answer main effect (pp)	+34.54	+9.87
Reasoning main effect (pp)	+42.43	+41.45

Table 13: **Accuracy after a wrong source across intervention protocols (%).** Each row supplies a wrong Base-produced source under the named prompt construction and reports the final-answer accuracy of the generated continuation or A_2. All panels use the same 19 questions and eight matched seeds; differences arise from content edits and, for two-pass conditions, the repeated-question frame.

Panel	Base	CRT
Boundary, full wrong	11.84	27.63
Crossed factorial, both wrong	11.84	31.58
Simple knockout, full wrong	6.58	32.89
Context budget, intact	7.24	33.55
Open full turn	7.89	30.26

Table 14: **Same-turn prefix interventions and generated-output effects.** After the user question, the experiment writes an assistant prefix containing optional Base-produced reasoning (with answer-equivalent spans masked), a fixed tentative boxed answer, and "Checking the work:". The model continues from that prefix and generates its final answer. Entries are changes in final-answer accuracy. The first two rows compare the complete correct or wrong reasoning prefix against an answer-only prefix with the same tentative answer. The third replaces only the last wrong reasoning step with a length-matched step from the correct trace and compares against the unchanged wrong prefix. Interaction is CRT minus Base; each cell uses 19 questions and 32 matched rollouts per question.

Prefix intervention	Base Δ acc.	CRT Δ acc.	CRT $-$ Base
Add complete wrong reasoning	-25.99	-55.10	$\mathbf{-29.11}$
Add complete correct reasoning	$+19.57$	$+27.14$	$+7.57$
Replace last wrong step with correct	$+13.49$	$+14.80$	$+1.32$

D Base-Only Forced Scoring and History-Component Probes

Both probes use JustRL-Nemotron-1.5B on held-out JustRL training-pool cases. Extracted answers use the same mathematical grader as the main evaluation. They are Base-only, and their decoding settings differ from the 19-question panel, so these rows are not directly comparable with panel cells.

A: forced suffix scoring. Each `\boxed` candidate is force-scored, without sampling, after the question-only (regular) and $Q_1/A_1/Q_2$ (replay) prompts; the metric is length-normalized mean logprob in nats. Of 64 sampled cases, 59 had a wrong extractable A_1 and 5 were skipped (no boxed answer). Table 15 reports the contrasts in the two scoring directions.

Table 15: **Does replay favor the model's previous wrong answer?** Here A_1 is Base's previous wrong answer and GT is the correct answer. For 59 cases, we force-score either output suffix without sampling after a regular input Q or replay input (Q, A_1, Q). The first two rows fix the output and compare inputs; the last two fix the input and compare outputs. Values are length-normalized mean logprob differences in nats.

Held fixed	Logprob comparison	Value
Output suffix A_1	Replay input $-$ regular input	$+4.452$
Output suffix GT	Replay input $-$ regular input	$+3.863$
Replay input	A_1 suffix $-$ GT suffix	$+0.756$
Regular input	A_1 suffix $-$ GT suffix	$+0.167$

E Attention Probe on the Question–Answer–Question Context

Setup. We compare Base against CRT on the JustRL family inside the fixed (x, y, x) context: question Q_1, one Base-generated response A_1, and a second question copy Q_2, followed by model generation A_2. One case without a complete reasoning block is excluded, leaving 31 paired cases. Query positions are an evenly spaced subset (at most 64 per case) of the frozen Base-derived low-gap token positions inside the A_2 reasoning block.

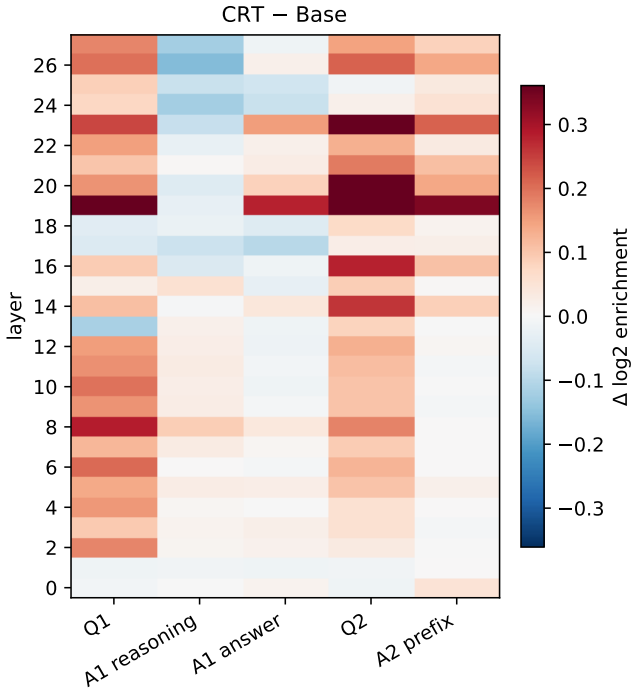


Figure 7: **Layer-wise attention change from A2 low-gap queries.** LowGap-SFT minus Base log2 attention enrichment across the last 8 layers and five key regions; the layer-averaged paired-case point estimates are in Table 16.

For each query token we read attention from the last 8 layers and all heads and partition the strictly earlier context into five key regions.

Region definition. The reasoning/answer boundary is taken from the model's literal `<think>` and `</think>` markers. *A1 reasoning* contains every token strictly inside the A_1 reasoning block—the Base-selected low-gap tokens and the remaining reasoning tokens merged into one region; *A1 answer* contains every token after the closing marker up to the end of A_1. *Q1* and *Q2* are the two question copies, and *A2 prefix* contains current-response tokens before each query. Only causally visible directions are measured.

Metrics. For region r with $|r|$ keys, attention mass is $M_r = \sum_{j \in r} a_{qj}$, per-key density is $D_r = M_r/|r|$, and log2 enrichment is $\log_2(D_r T)$ with T the number of causally visible keys; for the merged region the log2 enrichment is the key-count-weighted mean of the two subregions. Values are averaged over query positions, heads, and the last 8 layers at the case level. CRT−Base deltas are averaged over paired cases; point estimates are reported.

Table 16: **Attention changes from A2 low-gap queries.** In the fixed (Q_1, A_1, Q_2) input, rows partition the causally visible keys attended to by low-gap query tokens in the generated A_2. Entries are CRT minus Base, averaged over 31 paired cases, sampled query positions, all heads, and the last 8 layers.

Key region	Δ mass (pp)	Δ/key ($\times 10^5$)	Δ log2 enr.
Q1	+0.154	+1.998	+0.149
A1 reasoning	−1.560	−0.203	−0.080
A1 answer	−0.064	−0.117	+0.016
Q2	+0.039	+0.441	+0.175
A2 prefix	+1.654	+0.777	+0.100

Observed pattern. The largest absolute change is a withdrawal from A1 reasoning: -1.56 pp of attention mass, mirrored by $+1.65$ pp onto the A2 prefix and small increases on both question copies. The A1 answer shows little per-token change (Δ log2 enrichment $+0.016$), with only a small absolute mass decrease.

Low-gap versus high-gap query positions. The same paired cases also include queries from the opposite gap tail (high-gap positions within the same A_2 reasoning block); nothing else changes. Table 17 reports attention mass per key region, averaged over the last 8 layers.

Table 17: **Attention mass (%) by query gap tail.** For the same 31 paired $(Q_1, A_1, Q_2) \to A_2$ cases as Table 16, rows select low- or high-gap query positions in A_2 and columns partition their causally visible keys. Values average over query positions, heads, and the last 8 layers. *A1 low-gap* is the Base-selected subset of A_1 reasoning; *A1 think* is the remainder.

Model	Query	Q1	A1 low-gap	A1 think	A1 answer	Q2	A2 prefix
Base	Low gap	1.12	2.84	18.37	1.64	0.73	45.07
	High gap	0.89	2.24	22.73	1.58	0.52	40.10
CRT	Low gap	1.27	2.58	17.08	1.57	0.77	46.68
	High gap	1.04	2.11	21.71	1.54	0.56	41.96

Within Base, high-gap queries place $+4.36$ pp more mass on the remaining A1 reasoning and -4.97 pp less on the A2 prefix than low-gap queries: the two gap tails have a distinct attention signature before any training. The CRT$-$Base shift has the same direction for both query types—A1 reasoning $-1.29/-1.02$ pp, A2 prefix $+1.61/+1.86$ pp—and the within-model high-minus-low contrast is similar after training (A1 reasoning $+4.36$ vs. $+4.63$ pp; A2 prefix -4.97 vs. -4.72 pp). Like the rest of this section this is correlational: the gap tail and attention allocation co-vary, which does not show that high-gap tokens causally carry the copying. These correlations localize attention shifts but do not establish causal information flow; the probe covers one JustRL checkpoint and 31 cases.