SELF-TIDY: SUPPRESSING SELF-RESOLVED DELIBERATION VIA SINGLE-STAGE REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Large reasoning models often revisit intermediate conclusions through verification, recomputation, or backtracking, producing unnecessarily long trajectories. Existing length-control methods typically address this inefficiency through response-level constraints, without explicitly accounting for the utility of individual reasoning sentences. To selectively suppress low-value reasoning, we introduce **Self-Tidy**, a reinforcement-learning method that operates within the standard rollout-to-update process. Self-Tidy scores sentence-level predictive-entropy excursions and subsequent recovery to identify candidate self-resolved deliberation. It withholds positive reinforcement from selected spans while retaining their negative learning signals and preserving the complete sampled trajectories, requiring neither external process supervision nor a separate trace-compression stage. Across three model settings and five mathematical reasoning benchmarks, Self-Tidy reduces mean generated response length by 34.6%–52.2% while maintaining or improving average answer accuracy. These results show that Self-Tidy achieves a competitive balance between answer accuracy and response length without external process supervision or a separate compression pipeline.

1 INTRODUCTION

Explicit intermediate reasoning has improved language-model performance on mathematical and other multi-step tasks (Wei et al., 2022; Wang et al., 2023). Large reasoning models extend this paradigm through reinforcement learning and longer sequences of computation, verification, and self-correction (DeepSeek-AI, 2025). Even after reaching a viable solution, a model may rederive intermediate results, reconfirm established conclusions, or repeatedly backtrack without changing its final answer (Aggarwal & Welleck, 2025; Yi & Wang, 2025; Su & Cardie, 2025). Such behavior may also be reinforced during post-training when successful trajectories receive positive learning signals without distinguishing the contributions of individual reasoning sentences. This motivates identifying and selectively suppressing low-value reasoning while preserving useful verification and correction.

Identifying such low-value reasoning is challenging because redundant confirmation and substantive correction can exhibit similar lexical cues and elevated predictive uncertainty. Existing methods offer limited support for making this distinction. Length-aware objectives, reasoning budgets, and early-exit methods control how much computation is generated (Aggarwal & Welleck, 2025; Yi & Wang, 2025; Su & Cardie, 2025; Ma et al., 2025; Liu et al., 2026), while compression methods shorten or transform reasoning traces (Xia et al., 2025; Shen et al., 2025; Zhang et al., 2025). These approaches do not directly resolve which individual reasoning sentences warrant suppression. Finer-grained methods estimate token- or step-level importance (Jang et al., 2025; Singh & Hakkani-T"ur, 2026; Li et al., 2026; Jiang et al., 2026; Yang et al., 2026; Samanta et al., 2026), but local importance estimates alone do not necessarily distinguish transient reconsideration from sustained correction. Examining how uncertainty evolves beyond the candidate span offers an additional signal for making this distinction.

We examine *reasoning forks*, points where a model checks or revises its derivation. Using discourse markers to locate candidate forks and text-only cues to group confirmation and correction events,

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.

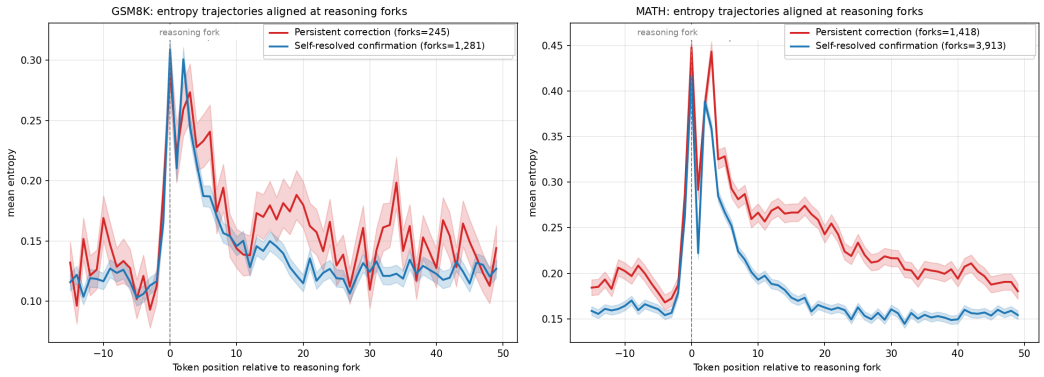


Figure 1: Fork-aligned predictive-entropy curves for text-heuristic confirmation and correction groups on GSM8K and MATH. Both groups show elevated uncertainty near the fork, while correction maintains higher post-fork entropy. Shaded bands are inherited from the original visualization; Section 2.1 reports the reproducible full-sample analysis.

we compare their aligned predictive-entropy profiles on GSM8K and MATH. Both groups show elevated uncertainty near the fork, but the confirmation group exhibits lower uncertainty afterward (Figure 1); a full-sample reanalysis supports this difference (Section 2.1). This motivates entropy recovery as a heuristic signal of *self-resolved deliberation*, where uncertainty rises and subsequently dissipates.

Motivated by this pattern, we introduce **Self-Tidy**, a reinforcement-learning method that discourages candidate low-value reasoning through selective credit assignment. Self-Tidy localizes sentence spans from the strength and subsequent recovery of their predictive-entropy excursions, then masks selected tokens from the policy loss only in positive-advantage trajectories. Non-positive-advantage trajectories remain unmasked, preserving negative learning signals when the advantage is negative. Because the sampled trajectories remain unchanged, each selected span continues to condition its suffix. Localization and credit masking occur within each standard rollout-to-update cycle, requiring neither external process supervision nor a separate compression stage.

We make three contributions.

- We characterize a fork-aligned entropy pattern in diagnostically labeled confirmation and correction events. Uncertainty rises near both types of reasoning forks but recovers more rapidly after confirmation, motivating post-fork recovery as a heuristic signal of self-resolved deliberation.
- We introduce Self-Tidy, which converts this signal into advantage-conditioned, sentence-level credit masking within each rollout-to-update cycle while preserving the complete sampled trajectory and its conditioning context.
- We evaluate Self-Tidy across three model settings and five mathematical reasoning benchmarks. It reduces mean generated response length by 34.6%–52.2% relative to the corresponding standard-prompt references while maintaining or improving macro accuracy.

2 METHOD

Self-Tidy changes token-level optimization credit without editing the sampled rollout. It scores sentences by the strength and recovery of their predictive-entropy excursions, selects a bounded set of candidates, and masks their positive credit during the GRPO update.

2.1 ENTROPY-GUIDED SELF-RESOLUTION LOCALIZATION

Reasoning forks are identified using predefined discourse markers. Each continuation receives a heuristic diagnostic label according to whether it preserves the preceding intermediate result or

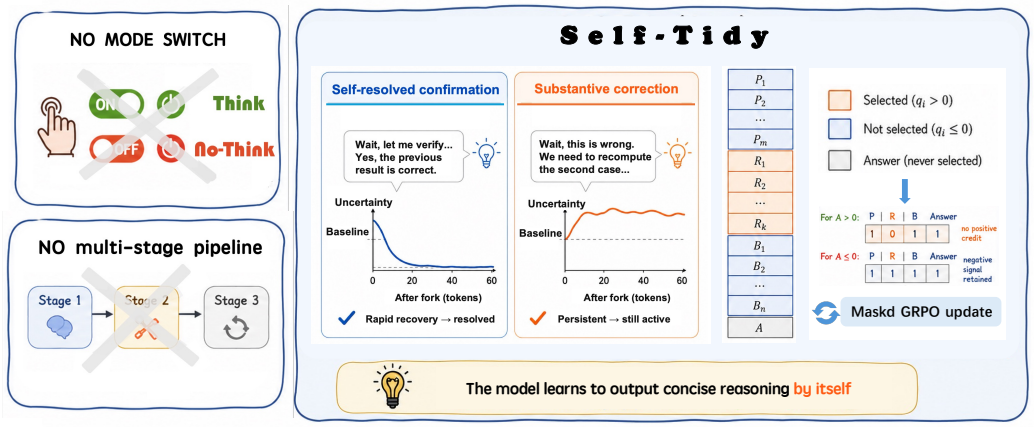


Figure 2: Overview of Self-Tidy. The method consists of sentence-level entropy scoring, budgeted span selection, and advantage-conditioned credit masking. Selected spans lose positive credit during the GRPO update while the sampled trajectory remains unchanged.

retracts and revises that result. The labels are assigned before entropy computation. Across the full Qwen3-8B analysis sets, the mean post-fork entropy above the pre-fork baseline is higher for correction than confirmation by 0.0252 nats on GSM8K and 0.0326 nats on MATH. This separation motivates post-fork entropy recovery as a localization signal, although lexical labels do not establish semantic dispensability. Appendix F.1 provides the complete labeling and statistical protocol.

Self-Tidy estimates entropy excursion and recovery directly from each sampled response.

Given a prompt x, the rollout policy generates

$$y = (y_1, \ldots, y_T) \sim \pi_{\theta_{\text{old}}}(\cdot \mid x). \tag{1}$$

For each response token, we compute the predictive entropy

$$H_t = -\sum_{v \in \mathcal{V}} \pi_{\theta_{\text{old}}}(v \mid x, y_{<t}) \log \pi_{\theta_{\text{old}}}(v \mid x, y_{<t}), \tag{2}$$

where $\mathcal{V}$ is the vocabulary.

The response is partitioned into complete sentences, $y = s_1 \oplus \cdots \oplus s_M$. For each sentence s_i, H_i^{own} denotes its mean token entropy and H_i^{post} denotes the mean entropy over a fixed following window. Response-relative normalization uses

$$b = Q_{0.20}(H), \qquad h = Q_{0.95}(H), \tag{3}$$

where $Q_p(H)$ is the p-th quantile of token entropy in the response.

The localization score combines the normalized excursion strength with its subsequent recovery.

$$g_i = \text{clip}\left(\frac{H_i^{\text{own}} - b}{h - b + \epsilon}, 0, 1\right), \tag{4}$$

$$r_i = \text{clip}\left(\frac{H_i^{\text{own}} - H_i^{\text{post}}}{H_i^{\text{own}} - b + \epsilon}, 0, 1\right), \tag{5}$$

$$q_i = g_i r_i. \tag{6}$$

The factor g_i measures excursion strength relative to the response's confident regime, while r_i measures the fraction of that excursion that dissipates afterward. Their product q_i is high only for pronounced excursions followed by substantial recovery.

Qualitative examples of the selector's decisions are provided in Appendix D.1.

2.2 BUDGETED SPAN SELECTION

A sentence is eligible when $q_i > 0$, it excludes the final answer, and it satisfies the context and minimum-length requirements. For each eligible sentence, we define

$$v_i = q_i \min\left(|s_i|, \ell_{\text{cap}}\right), \tag{7}$$

where ℓ_{cap} limits the influence of unusually long sentences. We greedily select candidates by v_i under the response-level budget

$$\sum_{s_i \in \mathcal{S}(y)} |s_i| \leq \beta T, \tag{8}$$

where β is the maximum fraction of response tokens whose credit may be modified. Let $c_t = 1$ when token y_t belongs to a selected sentence and $c_t = 0$ otherwise. Self-Tidy abstains when no sentence is eligible.

2.3 ADVANTAGE-CONDITIONED SEQUENCE-PRESERVING CREDIT ASSIGNMENT

For response i, let $A^{(i)}$ denote the standard GRPO advantage (Shao et al., 2024). The token-level credit mask is

$$m_t^{(i)} = 1 - c_t^{(i)} \mathbf{1}\left[A^{(i)} > 0\right]. \tag{9}$$

Selected tokens are excluded from the policy objective when $A^{(i)} > 0$. When $A^{(i)} \leq 0$, the mask remains one for every token, retaining negative updates for trajectories with $A^{(i)} < 0$.

With the standard PPO/GRPO clipping and regularization terms omitted for clarity (Schulman et al., 2017; Shao et al., 2024), the token-mean objective is

$$\mathcal{L}_{\text{ST}}(\theta) = -\frac{\sum_{i=1}^{G} \sum_{t=1}^{T_i} m_t^{(i)} A^{(i)} \log \pi_\theta\left(y_t^{(i)} \mid x, y_{<t}^{(i)}\right)}{\sum_{i=1}^{G} \sum_{t=1}^{T_i} m_t^{(i)}}. \tag{10}$$

The denominator normalizes over unmasked tokens and avoids additional response-level rescaling. Credit masking leaves the sampled context unchanged. For a trajectory $P \oplus R \oplus B$ with selected span R, the suffix B remains conditioned on $P \oplus R$. The original rollout states, rewards, and likelihood context are therefore preserved. Localization and credit assignment are completed within each standard rollout-to-update cycle without trajectory rewriting.

3 EXPERIMENTS

3.1 EXPERIMENTAL SETUP

Models and training. We evaluate Self-Tidy on DeepScaleR-1.5B-Preview and the 1.5B and 7B variants of DeepSeek-R1-Distill-Qwen (DeepSeek-AI, 2025). All models are trained with GRPO using outcome-based rewards. Unless otherwise specified, Self-Tidy uses response-relative entropy quantiles of 0.20 and 0.95, a 30-token post-sentence window, a minimum sentence length of three tokens, and a credit budget of 15% of the generated response. The objective uses token-mean loss aggregation.

Benchmarks and evaluation. We evaluate on MATH, Minerva Math, OlympiadBench, AIME 2024, and AMC 2023. For each problem, we sample 16 completions using temperature 0.6, top-p 0.95, a maximum response length of 32,768 tokens. Comparisons use the same prompt format, decoding configuration, answer extractor, and evaluator whenever matched results are available.

Metrics. We report avg@16 accuracy, mean generated response length, and Efficiency-F1 (E-F1). avg@16 averages correctness over 16 sampled completions per problem. Generated length includes the complete response and serves as the measure of inference efficiency, whereas credited length excludes tokens masked during training. Following AutoThink (Tu et al., 2025), E-F1 combines accuracy and length gains relative to the standard-prompt and no-thinking references. Detailed definitions are provided in Appendix C.3.

Method	MATH		Minerva		Olympiad		AIME24		AMC23		Average		Trade-off
	Acc. ↑	Tok. ↓	Acc. ↑	Tok. ↓	Acc. ↑	Tok. ↓	Acc. ↑	Tok. ↓	Acc. ↑	Tok. ↓	Acc. ↑	Tok. ↓	E-F1 (%) ↑
DeepScaleR-1.5B													
Standard prompt	87.60	3171	30.70	4948	50.00	5967	42.30	9326	72.80	5675	56.68	5817	0.0
No-thinking	72.55	825	19.37	790	35.96	1641	17.71	3076	53.59	1213	39.84	1509	0.0
Ellipsis	85.90	2890	28.90	4748	48.10	5416	42.10	9408	72.00	5095	55.40	5511	0.0
ThinkPrune-iter-2K	86.30	1838	30.70	2414	48.30	3254	38.70	5328	72.20	3166	55.24	3200	0.0
ThinkPrune-4K	86.50	2221	30.60	3039	48.50	4061	36.50	6624	71.80	3868	54.78	3962	0.0
AutoThink-S3	85.10	1897	30.50	3834	49.00	5005	41.90	9033	71.90	4696	55.68	4893	0.0
Self-Tidy	86.34	2202	31.18	2869	48.06	4050	39.17	6336	81.25	3563	**57.20**	3804	**5.8**
DeepSeek-R1-Distill-Qwen-1.5B													
Standard prompt	83.10	5622	26.00	7688	43.70	11555	27.50	17322	62.50	10981	48.56	10633	0.0
No-thinking	67.96	732	17.21	604	30.60	1849	15.42	3640	49.69	1347	36.18	1634	0.0
Ellipsis	78.20	4194	21.90	4336	38.60	7752	25.20	13006	57.20	7980	44.22	7453	0.0
ShorterBetter	–	–	27.60	1147	38.40	1814	20.00	2703	56.60	1946	–	–	–
ThinkPrune-iter-2K	82.60	1927	28.10	2126	43.60	3683	26.70	5806	64.90	3300	49.18	3368	9.4
ThinkPrune-4K	83.50	2723	28.40	3375	43.40	5504	28.30	8072	65.40	5040	49.80	4942	17.3
SAT	82.40	3230	–	–	–	–	26.70	10893	68.30	6642	–	–	–
AutoThink-S3	84.00	2195	28.10	3212	44.80	5559	34.60	9514	67.00	5059	**51.70**	5107	**35.9**
Self-Tidy	83.26	2936	27.20	3058	42.97	5728	29.79	9133	71.41	4535	50.93	5078	29.2
DeepSeek-R1-Distill-Qwen-7B													
Standard prompt	92.30	3928	37.60	5155	56.40	8815	52.70	13563	82.80	7613	64.36	7814	0.0
No-thinking	76.91	616	21.69	513	38.81	1341	17.71	2527	55.94	872	42.21	1174	0.0
Ellipsis	91.80	3752	37.60	4778	56.50	8643	51.30	13532	80.90	7616	63.62	7664	0.0
Concise-RL	90.30	2041	–	–	–	–	51.70	6632	–	–	–	–	–
ShorterBetter	–	–	44.10	1341	50.70	3410	53.30	5288	75.90	2580	–	–	–
AutoThink-S3	91.20	2146	38.20	2838	56.40	5498	54.80	8051	83.30	4645	64.78	4635	3.6
Self-Tidy	91.57	2587	38.70	3370	54.81	5614	52.50	8804	88.12	4081	**65.14**	4891	**6.5**

Table 1: Accuracy–length comparison across three model settings. Self-Tidy follows our evaluation protocol, whereas prior-method results are source reported. Tok. denotes mean generated response length. Average and E-F1 aggregate all five benchmarks. Cross-method comparisons are descriptive because evaluation protocols differ. Bold marks the best Average Acc. and E-F1 values within each model block, including ties.

3.2 Main Results

Table 1 compares Self-Tidy with prompting, pruning, and reinforcement-learning baselines across three model settings. Relative to the corresponding standard-prompt checkpoints, Self-Tidy consistently shifts the models toward shorter responses while preserving or improving average accuracy. Generated length decreases by 34.6%–52.2%, accompanied by macro-accuracy gains of 0.52–2.37 points. Self-Tidy therefore occupies a competitive accuracy–length region across model families and scales using a sequence-preserving, single-stage credit intervention. Because prior methods use heterogeneous evaluation protocols, cross-method results indicate comparative positioning, while the matched ablations below provide the controlled evidence for selective credit assignment.

Figure 3 shows the benchmark-level accuracy–length results from Table 1. Each point uses the method's own mean generated response length and accuracy. Arrows connect standard prompting to Self-Tidy and no-thinking on the same benchmark. For these within-benchmark comparisons, movement toward the upper left indicates fewer generated tokens and higher accuracy; movement toward the lower left indicates a reduction in length accompanied by an accuracy loss.

Self-Tidy reduces generated length on all five benchmarks in each model setting, with benchmark-dependent accuracy changes. No-thinking yields larger length reductions but lower accuracy than standard prompting on every displayed benchmark. The paired points expose these trade-offs without averaging away differences across benchmarks. As in Table 1, comparisons involving heterogeneous evaluation protocols are descriptive and do not establish statistical significance or per-problem adaptive compute allocation.

Additional three-seed results on DeepScaleR-1.5B-Preview are reported in Table S1 in the appendix.

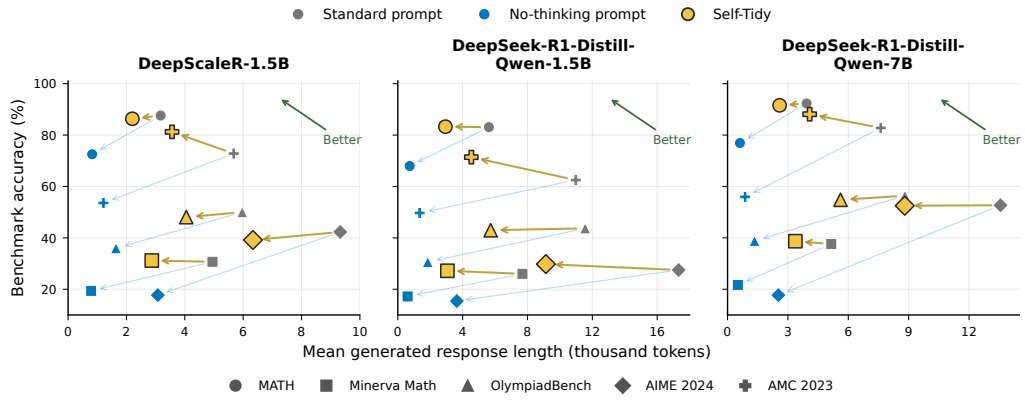


Figure 3: Benchmark-level accuracy–efficiency trade-offs across three models, using the results in Table 1. Colors denote methods and marker shapes denote benchmarks. Arrows connect standard prompting to each alternative on the same benchmark; movement toward the upper left indicates fewer generated tokens and higher accuracy. Each point reports the method's own mean generated response length and accuracy.

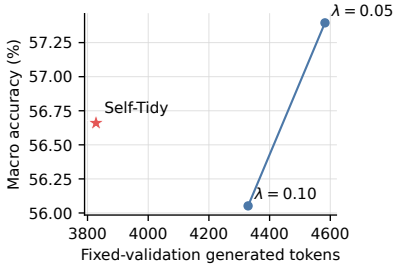


Figure 4: Accuracy–length comparison on DeepScaleR-1.5B-Preview. Self-Tidy is shorter than both global length-penalty settings and achieves intermediate accuracy.

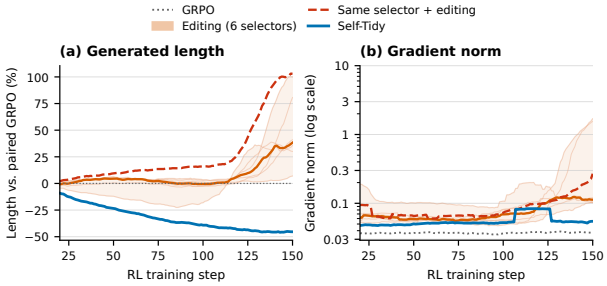


Figure 5: Trace editing versus credit-space localization under paired batch order. Trace-editing variants increase in gradient norm and generated length, whereas Self-Tidy remains comparatively stable. Curves show 20-step moving averages; differing loss aggregation makes this comparison diagnostic rather than causal.

3.3 Design Comparisons and Ablations

Global length penalty. Figure 4 compares Self-Tidy with global response-level length regularization on **DeepScaleR-1.5B-Preview**. Self-Tidy lies in a similar length range to the length-penalty runs while maintaining a competitive accuracy–length trade-off. Relative to the $\lambda = 0.10$ length penalty, Self-Tidy improves macro accuracy by 0.15 points while using 4.2% more generated tokens. The $\lambda = 0.05$ setting obtains 0.23 points higher accuracy and uses 1.5% more tokens than Self-Tidy. This pattern supports a narrower mechanism-level conclusion: localized credit can reach a comparable operating region without an explicit response-level length reward, while global length penalties remain sensitive to their coefficient.

The full numerical comparison is reported in Table S2; these nearby operating points do not establish superiority at equal generated length.

Trace editing. Self-Tidy modifies token credit while preserving the sampled rollout. Figure 5 contrasts this design with directly deleting the same types of localized spans before optimization. Under paired batch order, trace-editing variants exhibit increasing gradient norms followed by longer generated responses, whereas the credit-space variant remains comparatively stable over the observed

Variant	Val. (%) ↑	Gen. ↓	Cred.	Mask (%)
Matched GRPO	61.36	5807.6	5807.6	0.00
Random localization	60.84	5980.2	5777.9	3.68
Self-Tidy	58.76	4005.1	3802.0	4.94
Budget 30%	58.04	3887.9	3588.7	8.05
Symmetric mask	56.86	11501.6	9777.6	14.97

Table 2: localization and credit ablations on DeepScaleR-1.5B. Val. denotes validation reward; Gen. and Cred. denote the generated and credited lengths of the training batch. Mask is the realized fraction of generated tokens excluded from positive credit. Each row corresponds to one run.

training interval. Because the runs use different loss-aggregation rules, this comparison is diagnostic rather than a controlled causal ablation.

Localization and credit assignment. We ablate entropy-based localization, the credit budget, and advantage-conditioned masking. Matched GRPO disables masking, Random localization selects eligible sentences independently of their scores, Budget 30% doubles the credit budget, and Symmetric masking applies the mask regardless of advantage sign. All variants share the model, data order, optimizer, sampling configuration, and loss aggregation, and are compared at the same training step. Because each variant has one run, the results are descriptive. Table 2 reports validation reward together with generated and credited length, of which only generated length reflects the policy's actual output.

These ablations clarify how Self-Tidy produces its shortening effect. Random localization leaves generated behavior close to matched GRPO, indicating that masking eligible sentences alone is insufficient without entropy-based selection. The default selector yields a distinct shortening trajectory, while the larger budget strengthens this effect at the expected cost in validation reward, making the default budget the more conservative operating point. Symmetric masking degrades both response length and reward, consistent with the importance of retaining negative updates. Its larger realized masking rate makes this a directional sensitivity result rather than a gate-only causal estimate. Together, the variants support targeted localization and advantage-conditioned credit assignment as the components that shape the accuracy–length trade-off.

Benchmark-level validation results and training-length evolution are reported in Tables S4 and S5 in the appendix.

Additional training dynamics for DeepSeek-R1-Distill-Qwen-7B are reported in Figure S1.

3.4 Training Efficiency

Internal training-cost comparison. Self-Tidy increases per-token cost but shortens rollouts enough to reduce mean step time. Table 3 compares matched no-mask and Self-Tidy runs after ten warm-up steps on eight NVIDIA H200 GPUs.

Variant	Sec./step ↓	Tok./s/GPU ↑	Peak mem. ↓	Cost/tok.
No mask	146.6±15.0	2460±221	91.46	–
Self-Tidy	117.6±19.7	2131±316	96.76	+15.4%

Table 3: Training efficiency on DeepScaleR-1.5B-Preview after ten warm-up steps. Time and throughput are mean±standard deviation over 90 steps, and peak memory is measured per GPU in GB.

Self-Tidy reduces generated training length by 31.6% and mean step time by 19.8%, despite lower token throughput and a 5.8% increase in peak memory. The 15.4% cost increase measures the full pipeline because the entropy computation is not timed separately.

Cross-method context. Beyond accuracy and output length, we examine the training cost required to obtain the reported models. Table 4 reports 1.62, 2.82, and 4.99 hours of logged training time for

Self-Tidy across the three model settings. Published costs for AutoThink and ThinkPrune provide context for multi-stage and iterative training, while ShorterBetter provides a single-stage reference. Together with the accuracy–length results, these measurements show that Self-Tidy reaches competitive operating points through a single training stage. Cross-paper resource figures are descriptive because hardware, evaluation protocols, and cost accounting differ.

Table 4: Reported training costs across methods. Corresponding accuracy and output-length results are presented in Table 1.

Base model	Method	Stages	Time (h)↓	GPU hours↓
DeepScaleR-1.5B	AutoThink	3	$\leq 24^{\dagger}$	–
	Self-Tidy	1	1.62	12.96
R1-Distill-1.5B	ShorterBetter	1	≈ 16	≈ 64
	ThinkPrune-iter-2K	Iterative	–	$\approx 400^{\ddagger}$
	AutoThink	3	$\leq 24^{\dagger}$	$\approx 700^{\ddagger}$
	Self-Tidy	1	2.82	22.56
R1-Distill-7B	ShorterBetter	1	< 12	< 96
	AutoThink	3	$\leq 60^{\dagger}$	–
	Self-Tidy	1	4.99	39.90

R1-Distill denotes DeepSeek-R1-Distill-Qwen. Our times sum logged training-step timers to the reported checkpoints and exclude validation. GPU hours equal time multiplied by GPU count. † Author-reported AutoThink bounds. ‡ Estimates from Tu et al. (2025). ShorterBetter figures are source reported. Dashes denote unavailable values, and pre-existing base-model training is excluded.

4 Related Work

4.1 Efficient Reasoning and Length Control

Chain-of-thought prompting and sampling-based aggregation establish the value of explicit, diverse reasoning trajectories (Wei et al., 2022; Wang et al., 2023). Response-level efficiency objectives encourage correct solutions that use fewer tokens through length penalties, adaptive rewards, or controllable reasoning budgets (Aggarwal & Welleck, 2025; Yi & Wang, 2025; Su & Cardie, 2025; Ma et al., 2025). Early-exit and adaptive-decoding methods estimate when additional reasoning is unlikely to be useful (Liu et al., 2026). These approaches regulate the amount of computation allocated to a response. Self-Tidy contributes a local learning signal that identifies resolved deliberative behavior and suppresses its positive credit within a successful rollout.

4.2 Chain-of-Thought Compression and Pruning

Compression methods shorten explicit reasoning through token skipping, distillation, or latent representations (Xia et al., 2025; Shen et al., 2025; Zhang et al., 2025). A related family estimates removable reasoning units using answer likelihood or functional importance (Jang et al., 2025; Singh & Hakkani-T"ur, 2026). Self-Tidy characterizes each candidate through post-fork uncertainty dynamics and applies the localization decision within the current rollout-to-update cycle. The original training trace remains intact throughout the update.

4.3 Fine-Grained Credit in Reasoning RL

Fine-grained reasoning objectives assign outcome information selectively across tokens. Existing approaches estimate marginal token utility, focus updates on uncertain bottlenecks, generate targeted interventions, or resample suffixes from intermediate states (Li et al., 2026; Jiang et al., 2026; Yang et al., 2026; Samanta et al., 2026). Dynamic rollout editing modifies sampled reasoning before learning (Wei et al., 2026). Self-Tidy contributes a self-resolution signal within a single-stage, policy-conditioned training procedure. An advantage-conditioned mask transfers this signal into optimization while preserving the sampled context.

5 LIMITATIONS

Self-Tidy uses predictive entropy as a behavioral signal rather than a semantic test of redundancy. A rapidly resolving excursion can still support useful reasoning, whereas some redundant reasoning may remain low entropy. The selector also depends on sentence boundaries and a post-sentence context window, which may not align with every reasoning unit and provides limited evidence for spans near the end of a response. Computing full-vocabulary entropy adds training overhead, and response-relative thresholds can remain sensitive to distribution shift, although the credit budget bounds the intervention dose.

Our evidence is limited to verifiable mathematical reasoning and models up to 7B. Extending the method to open-ended reasoning, larger models, and semantically grounded redundancy measures remains future work.

6 CONCLUSION

We introduced Self-Tidy, a single-stage reinforcement-learning method for discouraging candidate self-resolved deliberation. Self-Tidy uses sentence-level entropy excursion and recovery to localize candidate spans, then withholds their positive credit while preserving the sampled trajectory and negative learning signals.

Across three model settings and five mathematical reasoning benchmarks, Self-Tidy consistently shifts the accuracy–length trade-off toward shorter generated responses relative to standard prompting while maintaining or improving macro accuracy. The controlled ablations further show that targeted localization and advantage-conditioned masking both contribute to this effect. Self-Tidy therefore provides a sequence-preserving approach to reducing candidate low-value reasoning without external process supervision or a separate compression stage. Its conclusions are limited to verifiable mathematical reasoning and the evaluated model scales.

AI USE STATEMENT

Generative AI tools were used to assist with literature discovery and organization, manuscript drafting and language editing, software implementation and debugging, experiment configuration, and the organization and interpretation of empirical results. The authors verified the cited sources and reviewed all AI-assisted text, code, analyses, and claims. These tools were not used to generate synthetic datasets or to prove mathematical claims. The authors take responsibility for the final content of this work.

REFERENCES

Pranjal Aggarwal and Sean Welleck. L1: Controlling how long a reasoning model thinks with reinforcement learning, 2025.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Joonwon Jang, Jaehee Kim, Wonbin Kweon, Seonghyeon Lee, and Hwanjo Yu. Verbosity-aware rationale reduction: Sentence-level rationale reduction for efficient and effective reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20769–20784. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.1068. URL https://aclanthology.org/2025.findings-acl.1068/.

Shuyang Jiang, Yuhao Wang, Ya Zhang, Yanfeng Wang, and Yu Wang. MINER: Mining intrinsic mastery for data-efficient RL in large reasoning models, 2026.

Jiawei Li, Yang Gao, Huashan Sun, and Chong Feng. Think better, not longer: Token-level marginal utility for efficient reasoning in large reasoning models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics*, pp. 30052–30063. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1386.

Fan Liu, Yanhao Wang, Min Zhang, Zhikang Chen, Zeyuan Li, Lewei He, and Jiahui Pan. Think faster than words: Efficient LLM chain-of-thought reasoning via dynamic shortcut decoding. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics*, pp. 28825–28836. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1330.

Xinyin Ma, Guangnian Wan, Runpeng Yu, Gongfan Fang, and Xinchao Wang. CoT-Valve: Length-compressible chain-of-thought tuning, 2025.

Ankur Samanta, Akshayaa Magesh, Ayush Jain, Youliang Yu, Daniel Jiang, Kavosh Asadi, Kaveh Hassani, Paul Sajda, Jalaj Bhandari, and Yonathan Efroni. Credit assignment with resets in language model reasoning, 2026.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

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, 2024. URL https://arxiv.org/abs/2402.03300.

Zhenyi Shen, Hanqi Yan, Linhai Zhang, Zhanghao Hu, Yali Du, and Yulan He. CODI: Compressing chain-of-thought into continuous space via self-distillation, 2025.

Janvijay Singh and Dilek Hakkani-T"ur. Do LLMs encode functional importance of reasoning tokens?, 2026.

Jinyan Su and Claire Cardie. Thinking fast and right: Balancing accuracy and reasoning length with adaptive rewards, 2025.

Songjun Tu, Jiahao Lin, Qichao Zhang, Xiangyu Tian, Linjing Li, Xiangyuan Lan, and Dongbin Zhao. Learning when to think: Shaping adaptive reasoning in R1-style models via multi-stage RL, 2025. URL https://arxiv.org/abs/2505.10832.

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 *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

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

Zihao Wei, Wenjie Shi, Liang Pang, Jingcheng Deng, Shicheng Xu, Shasha Guo, Zenghao Duan, Jiahao Liu, Jingang Wang, Huawei Shen, and Xueqi Cheng. Dynamic rollout editing for reducing overthinking in RL-trained reasoning models, 2026.

Heming Xia, Yongqi Li, Chak Tou Leong, Wenjie Wang, and Wenjie Li. TokenSkip: Controllable chain-of-thought compression in LLMs, 2025.

Matthew Y. R. Yang, Hao Bai, Ian Wu, Gene Yang, Amrith Setlur, and Aviral Kumar. InT: Self-proposed interventions enable credit assignment in LLM reasoning, 2026.

Jingyang Yi and Jiazheng Wang. Shorterbetter: Guiding reasoning models to find optimal inference length for efficient reasoning, 2025.

Jintian Zhang, Yuqi Zhu, Mengshu Sun, Yujie Luo, Shuofei Qiao, Lun Du, Da Zheng, Huajun Chen, and Ningyu Zhang. LightThinker: Thinking step-by-step compression, 2025.

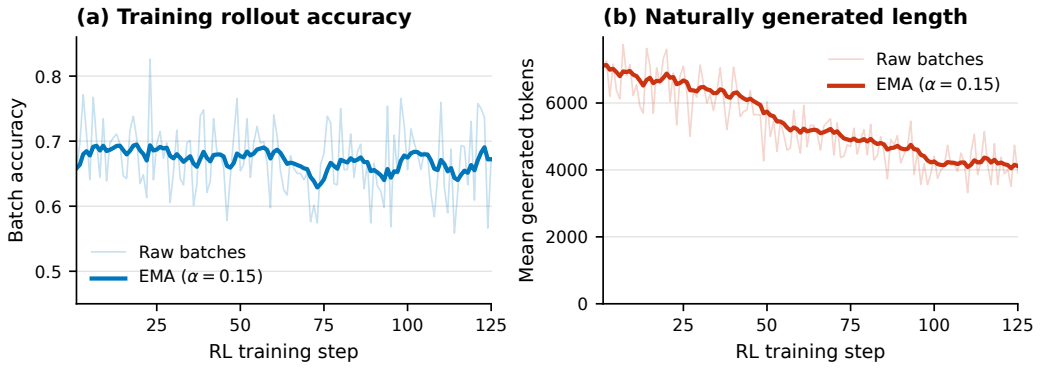


Figure S1: Training dynamics of Self-Tidy on DeepSeek-R1-Distill-Qwen-7B. The curve shows that the method maintains a stable training trajectory while reducing generated response length, indicating convergent behavior rather than unstable optimization.

A Extended Experimental Results

A.1 Multi-Seed Results on DeepScaleR-1.5B

Table S1 evaluates whether the Self-Tidy operating point is stable across seeds on **DeepScaleR-1.5B-Preview**. All fixed-validation results use the same checkpoint and the unweighted macro-average over the five main benchmarks: MATH, Minerva Math, OlympiadBench, AIME 2024, and AMC 2023. Fixed-val Gen. is the corresponding validation rollout length. Train Cred. and Mask are exact-step training-batch statistics and are reported separately to avoid conflating optimized credit with inference-time generation.

Variant	Acc. ↑	Gen. ↓	Cred. ↓	Mask
GRPO(No mask)	56.93	6153	5597	0.00
Self-Tidy(3 seeds)	$\mathbf{56.66 \pm .004}$	$\mathbf{3828 \pm 14}$	$\mathbf{2906 \pm 61}$	$\mathbf{4.43 \pm .62}$

Table S1: Multi-seed controlled results on DeepScaleR-1.5B-Preview. Self-Tidy reports the mean and variation across three seeds.

The multi-seed results show that Self-Tidy attains a consistent accuracy–length operating point, with limited variation in both macro accuracy and validation length. This stability suggests that the observed reduction in generated tokens is not attributable to a single favorable seed. Across repeated runs, Self-Tidy reproducibly enters a compressed generation regime while preserving a comparable accuracy level, and the reported accuracy change quantifies the trade-off associated with this more aggressive shortening.

A.2 Training Dynamics on DeepSeek-R1-Distill-Qwen-7B

Figure S1 reports the training dynamics of Self-Tidy on DeepSeek-R1-Distill-Qwen-7B. Training rollouts accuracy remains stable over training, while the generated response length decreases rather than exhibiting divergent growth. This behavior indicates that the credit-masking intervention yields a convergent optimization process in the larger 7B setting, supporting the view that Self-Tidy does not rely on unstable length suppression.

A.3 Length-Penalty Comparison Details

Table S2 compares Self-Tidy with the completed sweep of response-level length penalties near the same generation-length region.

Table S2 does not establish superiority over a global length penalty at equal length. Instead, it shows that Self-Tidy occupies a competitive region of the accuracy–length frontier that compared with

Algorithm 1 Self-Tidy Credit-Masking Update

Require: Rollout batch $\{(x^{(j)}, y^{(j)}, A^{(j)})\}_{j=1}^{G}$; budget β; post-window w; quantiles $p_{\mathrm{lo}}, p_{\mathrm{hi}}$; minimum length $\ell_{\min}$; length cap ℓ_{cap}

for $j = 1, \ldots, G$ **do**
  Compute token entropies $H_{1:T_j}$ and tokenizer-aligned sentence spans
  $b \leftarrow Q_{p_{\mathrm{lo}}}(H)$; $h \leftarrow Q_{p_{\mathrm{hi}}}(H)$; $\mathcal{C} \leftarrow \emptyset$
  for each non-answer sentence s_i with $|s_i| \geq \ell_{\min}$ and w following tokens **do**
    Compute H_i^{own} and H_i^{post}
    $q_i \leftarrow \mathrm{clip}\Big(\frac{H_i^{\mathrm{own}} - b}{h - b}, 0, 1\Big)\,\mathrm{clip}\Big(\frac{H_i^{\mathrm{own}} - H_i^{\mathrm{post}}}{H_i^{\mathrm{own}} - b}, 0, 1\Big)$
    if $q_i > 0$ **then**
      Add $(s_i, q_i \min(|s_i|, \ell_{\mathrm{cap}}))$ to $\mathcal{C}$
    end if
  end for
  Greedily select $\mathcal{S}_j$ by decreasing value subject to $\sum_{s_i \in \mathcal{S}_j} |s_i| \leq \beta T_j$
  $m_t^{(j)} \leftarrow 1 - \mathbf{1}[y_t^{(j)} \in \mathcal{S}_j]\mathbf{1}[A^{(j)} > 0]$
end for
Update θ with the token-mean GRPO objective using $m_t^{(j)}$
return π_θ; rollout sequences remain unchanged

Variant	Coef.	Acc. ↑	Gen. ↓	Gap
Self-Tidy	–	$\mathbf{56.66 \pm .004}$	$\mathbf{3828 \pm 14}$	–
Length penalty	0.10	56.51	3669	-4.2%
Length penalty	0.05	56.89	3885	$+1.5\%$

Table S2: Length-penalty sweep on DeepScaleR-1.5B-Preview near the Self-Tidy operating region. Gap reports relative generated length compared with Self-Tidy.

$\lambda = 0.10$, it is slightly longer but more accurate; compared with $\lambda = 0.05$, it is slightly shorter but less accurate. Self-Tidy reaches a stable compressed operating point without an explicit response-length reward or a tuned global length coefficient, while retaining an interpretable sentence-level credit signal.

B Self-Tidy Algorithm

B.1 Inputs and Update Semantics

Algorithm 1 makes the paper's central intervention explicit: the method changes which sampled tokens receive policy-gradient credit, rather than editing or shortening the sampled trajectory. This separation is important for interpreting the main results: any reduction in generated length is an emergent consequence of the update, not a length change injected into the training target. The entropy statistics and mask are computed from each sampled response, and the original token sequence and rollout log-probabilities are retained for the policy update.

B.2 Default Configuration

Self-Tidy may abstain when $\mathcal{C}$ is empty. In the reported default configuration, $(p_{\mathrm{lo}}, p_{\mathrm{hi}}) = (0.20, 0.95)$, $w = 30$ tokens, $\ell_{\min} = 3$ tokens, $\ell_{\mathrm{cap}} = 40$ tokens, $k_{\min} = 64$ kept tokens, and $\beta = 0.15$. Standard GRPO clipping and regularization terms remain unchanged and are omitted from the pseudocode. The implementation sets $q_i = 0$ when either denominator in Algorithm 1 is non-positive. If the entropy vector is missing or misaligned, or if applying the mask would leave fewer than $k_{\min}$ tokens, it falls back to an all-ones mask for that response.

Setting	Value
Training prompts per update	64
Rollouts per prompt	8
Maximum response length	32,768
Loss aggregation	Token mean
Entropy quantiles $(p_{\mathrm{lo}}, p_{\mathrm{hi}})$	$(0.20, 0.95)$
Post-sentence window	30 tokens
Minimum sentence length	3 tokens
Default credit budget β	0.15

Table S3: Configuration shared by the controlled DeepScaleR-1.5B runs.

C Experimental Details

C.1 Training and Method Configuration

Table S3 summarizes the settings shared by the controlled DeepScaleR-1.5B runs. The optimizer, data order, prompt format, reward, and sampling implementation are inherited from the matched GRPO configuration.

C.2 Controlled Variants

The ablations use DeepScaleR-1.5B-Preview, the same training data and batch order, token-mean loss aggregation, a train batch size of 64, eight rollouts per prompt, and a maximum response length of 32,768 tokens. Matched GRPO executes the detector for measurement only. Random localization preserves the positive-advantage gate and selects the same number of sentences from the scoreable sentence pool, independently of their localization score; thus the random control preserves the structural eligibility guards but does not require the selected sentences to have positive excursion scores. Self-Tidy ranks sentences by the entropy excursion–recovery score and masks selected spans only on positive-advantage trajectories. The 30% budget variant doubles the default response-level credit budget from $\beta = 0.15$ to $\beta = 0.30$, matching the budget definition above. Symmetric masking applies localized masks regardless of advantage sign.

C.3 Evaluation Protocol

For each problem, $\mathrm{avg@16}$ is the mean correctness over 16 sampled completions from one checkpoint. Benchmark accuracy averages this quantity over problems, and the reported Average is the unweighted mean across the five benchmarks. Response length is the mean number of generated tokens, including reasoning and the final answer. Generated length records the policy's complete output, while credited length excludes tokens masked from the training objective. Only generated length is used to assess inference efficiency.

Following AutoThink (Tu et al., 2025), E-F1 is computed from the five-benchmark Average accuracy $\bar{A}$ and token length $\bar{L}$. Let $\Delta_{\mathrm{acc}} = (\bar{A} - \bar{A}_{\mathrm{std}})/(\bar{A}_{\mathrm{std}} - \bar{A}_{\mathrm{no}})$ and $\Delta_{\mathrm{len}} = (\bar{L}_{\mathrm{std}} - \bar{L})/(\bar{L}_{\mathrm{std}} - \bar{L}_{\mathrm{no}})$, where std and no denote the standard-prompt and no-thinking references. We report $100 \cdot 2\Delta_{\mathrm{acc}}\Delta_{\mathrm{len}}/(\Delta_{\mathrm{acc}} + \Delta_{\mathrm{len}})$ when $\bar{A} > \bar{A}_{\mathrm{std}}$ and $\bar{L} < \bar{L}_{\mathrm{std}}$. Otherwise E-F1 is zero, and it is left undefined when results are unavailable for any of the five benchmarks.

All variants are compared at common training checkpoints. The multi-seed comparison in Table S1 is used to assess repeatability of the Self-Tidy operating point, while the length-penalty comparison is interpreted separately as a coefficient sweep. Validation is performed with four sampled completions per problem. Generated and credited lengths are exact-step training-batch statistics, whereas validation reward is computed by checkpoint evaluation. These values therefore answer different questions and should not be conflated. Controls and sensitivity variants remain single-seed and are interpreted descriptively unless otherwise stated.

C.4 Prompt Templates

The standard, no-thinking, and ellipsis prompting baselines follow the AutoThink baseline templates. The task question is supplied before the suffix shown below; `\n` denotes a newline. The standard prompt opens the reasoning span and lets the model continue normally:

```
<think>\n
```

The no-thinking prompt immediately closes the reasoning span with the fixed sentence used by the reference implementation:

```
<think> Okay, I think I have finished thinking.
</think>
```

For completeness, the ellipsis baseline inserts an open-ended continuation after the opening tag:

```
<think>\n...\n
```

These suffixes affect only the prompting baselines in the main comparison. Self-Tidy uses the standard rollout prompt and introduces no inference-time mode-selection instruction.

D Additional Ablations and Diagnostics

D.1 Qualitative Analysis of Final Selector Decisions

We qualitatively audit the final localization selector by replaying it on 120 MATH completions sampled from the DeepScaleR-1.5B Self-Tidy , with one completion per problem. Figure S2 shows four representative spans from responses that are correct under the paper's math evaluator. The examples are intended to illustrate selector behavior rather than estimate the prevalence or causal removability of each reasoning category.

Panels (a)–(c) illustrate distinct forms of already-resolved computation, whereas panel (d) shows the intended safeguard: a large uncertainty excursion alone is insufficient when uncertainty remains elevated during continued correction.

D.2 Benchmark-Level Ablation Results

The benchmark-level results in Table S4 show that the aggregate trade-off is not attributable to a single dataset. Relative to matched GRPO, Self-Tidy matches AIME24 and improves AMC23, while decreasing reward on MATH, Minerva, and OlympiadBench. Random localization remains close to matched GRPO in reward but leaves generated length unchanged, separating the shortening effect from outcome performance alone. Increasing the budget or removing the advantage gate generally worsens reward across the suite, consistent with the aggregate results in Table 2.

D.3 Training Evolution

Table S5 summarizes generated response length over successive twenty-step training intervals for the controlled variants. Each Gen. column is a mean over exact-step training batches, while Mask is the mean percentage of generated tokens excluded from positive credit through step 100. These trajectories characterize training dynamics and should not be interpreted as fixed-checkpoint validation accuracy.

Matched GRPO remains approximately flat, changing from 5406.5 tokens in steps 1–20 to 5223.3 in steps 81–100. Random localization is also flat, ending at 5389.7 tokens despite a realized mask rate (3.51%) close to Self-Tidy (3.55%). This matched-dose contrast indicates that the result is inconsistent with masking amount being the sole explanation. Self-Tidy instead decreases progressively from 4842.0 to 3136.2 tokens across the same intervals. Doubling the credit budget raises the realized mask rate to 7.14% and produces a slightly stronger reduction, ending at 2958.5 tokens, which is consistent with a dose-sensitive compression effect. Symmetric masking shows the opposite trajectory that length increases after step 20, peaks at 14740.0 tokens in steps 61–80, and remains at

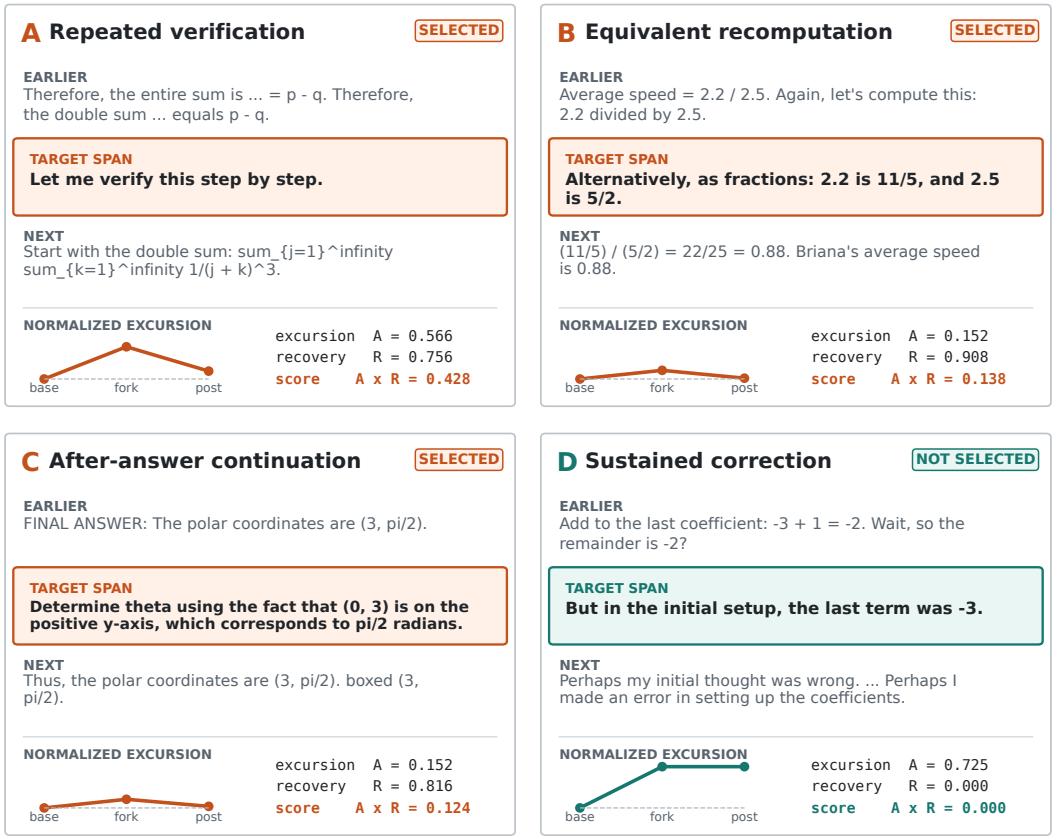


Figure S2: Representative decisions of the final Self-Tidy localization selector. Repeated verification, equivalent recomputation, and continuation after an explicit answer exhibit substantial post-span recovery and are selected. In contrast, the displayed span from a sustained correction is not selected because its uncertainty does not recover. The miniature diagrams encode normalized excursion amplitude A and residual excursion $A(1 - R)$; the localization score is AR. These examples isolate localization behavior: training-time positive-credit masking additionally requires positive sequence advantage, and the sampled response remains unchanged.

Variant	Overall	AIME24	AMC23	MATH	Minerva	Olympiad
Matched GRPO	61.36	39.17	78.12	88.70	32.35	49.96
Random localization	60.84	42.50	82.50	87.55	31.71	51.00
Self-Tidy	58.76	39.17	81.25	86.30	29.23	47.37
Budget 30%	58.04	38.33	80.62	85.60	29.14	47.41
Symmetric mask	56.86	33.33	74.38	83.70	28.49	44.89

Table S4: Step-50 validation reward by benchmark (%). Overall is the reward over the complete validation suite and is not the arithmetic mean of the five displayed benchmarks. Each result is based on one run with four completions per problem.

10896.3 in steps 81–100. Because this variant also has a larger realized mask rate, it is best interpreted as a sensitivity test of removing the advantage-conditioned gate rather than an equal-dose estimate of mask direction alone.

Variant	Gen. 1–20	Gen. 21–40	Gen. 41–60	Gen. 61–80	Gen. 81–100	Mask 1–100
Matched GRPO	5406.5	5492.5	5297.0	5346.8	5223.3	0.00
Random localization	5387.8	5477.6	5444.3	5552.9	5389.7	3.51
Self-Tidy	4842.0	4199.9	3714.5	3442.9	3136.2	3.55
Budget 30%	4829.3	4099.9	3666.1	3281.2	2958.5	7.14
Symmetric mask	5525.7	6594.1	11126.6	14740.0	10896.3	14.96

Table S5: Twenty-step means through step 100. Gen. is exact-step training-batch generated response length; Mask is the percentage of generated tokens excluded from positive credit, averaged over all 100 steps.

D.4 Optimization Diagnostics

Table S6 reports exact-step entropy, off-policy KL, and gradient norm for the same controlled variants.

Step	Variant	Entropy	Off-policy KL	Grad. norm
50	Matched GRPO	0.2087	0.001215	0.0492
50	Random localization	0.2059	0.001286	0.0411
50	Self-Tidy	0.1557	0.001012	0.0543
50	Budget 30%	0.1418	0.000994	0.0536
50	Symmetric mask	0.2112	0.001094	0.1671
100	Matched GRPO	0.2049	0.001154	0.0463
100	Random localization	0.2058	0.001215	0.0428
100	Self-Tidy	0.1428	0.000964	0.0586
100	Budget 30%	0.1120	0.000826	0.0546
100	Symmetric mask	0.3671	0.002877	0.6110

Table S6: Exact-step optimization diagnostics. Symmetric masking shows substantially larger entropy, KL, and gradient norm at step 100.

E Run Provenance and Limitations

E.1 Run Identifiers

The reported experiment identifiers are:

Matched GRPO	`deepscaler_1p5b_credit_none_v5`
Random localization	`deepscaler_1p5b_credit_random_v5`
Self-Tidy	`deepscaler_1p5b_entropy_credit_v5`
Budget 30%	`deepscaler_1p5b_credit_budget030_v5`
Symmetric mask	`deepscaler_1p5b_credit_sym_v5`

Complete checkpoints exist for all five variants.

The multi-seed and length-penalty tables use additional completed runs under the same DeepScaleR-1.5B-Preview evaluation protocol. These runs are reported only for repeatability and coefficient-sensitivity analyses, and are not treated as independent repetitions of the full ablation suite.

E.2 Resource Scope and Complementary Evidence

The controlled reinforcement-learning experiments focus on DeepScaleR-1.5B because full training sweeps at larger model scales require substantially greater computational resources. We therefore use the 1.5B controlled setting for causal ablations, repeatability checks, and length-penalty comparisons. To probe whether the motivating length and entropy phenomena also appear beyond this training scale, the following section provides complementary analyses on larger Qwen models, including an 8B setting. These analyses are used as motivation and boundary evidence rather than as substitutes for a full large-scale training ablation.

F Additional Motivation and Mechanism Evidence

This section provides complementary evidence for the mechanism assumptions used by Self-Tidy. The analyses are ordered from measurement validity to mechanism specificity: first, we audit whether raw response length is a reliable efficiency measure; second, we examine the accuracy–length surface after conditioning on correctness; third, we test whether entropy alone is sufficient

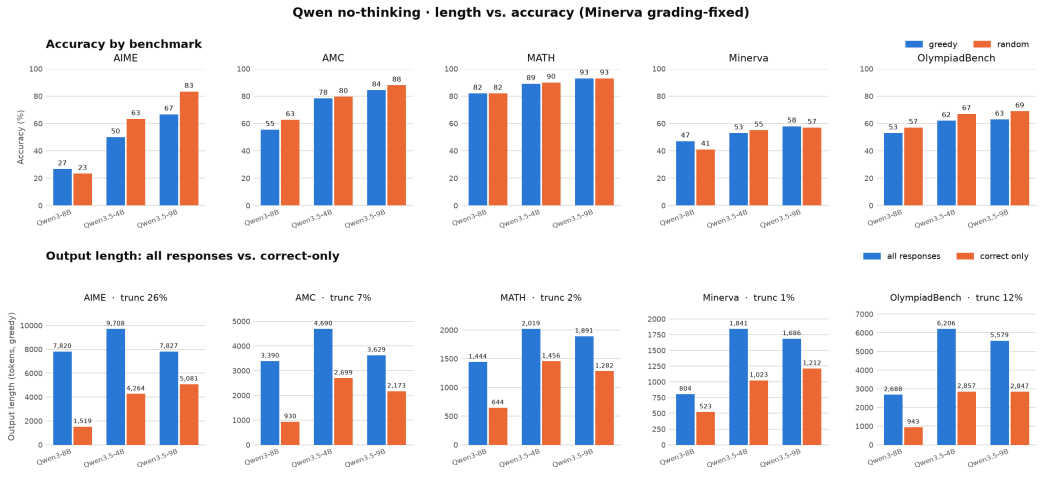


Figure S3: Accuracy and output-length audit under no-thinking decoding for Qwen3-8B, Qwen3.5-4B, and Qwen3.5-9B. The top row compares greedy and random-sampling accuracy by benchmark. The bottom row contrasts mean length over all responses with mean length over correct responses; panel titles also report truncation rates. Large all-response versus correct-only gaps expose length inflation from unsuccessful trajectories.

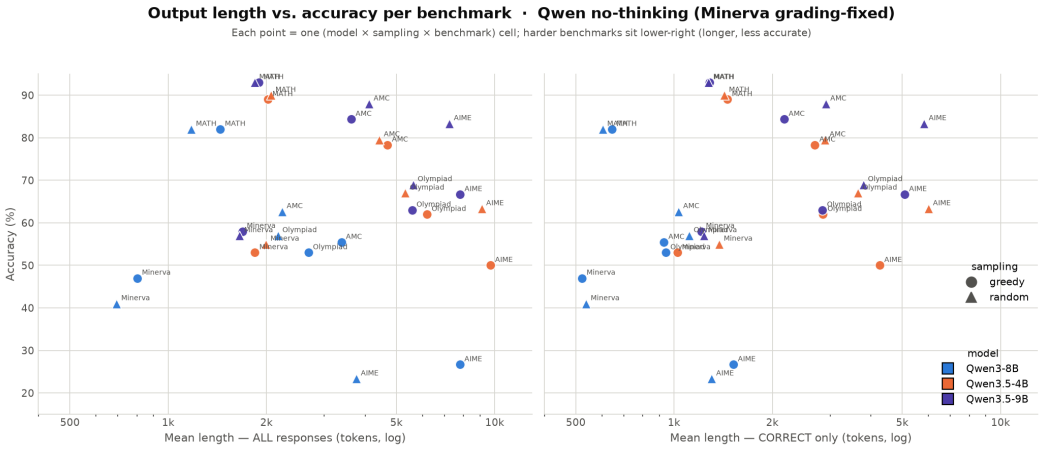


Figure S4: Accuracy versus mean output length for each model–sampling–benchmark cell under no-thinking decoding. The left panel uses all responses, whereas the right panel computes length only over correct responses. Color denotes model and marker shape denotes sampling strategy; the length axis is logarithmic. Correctness conditioning reduces a major source of length confounding but does not remove benchmark difficulty effects.

for identifying removable reasoning; finally, we characterize the sensitivity of the detector to threshold and distribution shifts.

F.1 Diagnostic Fork-Analysis Protocol

We locate candidate forks using discourse markers such as *wait*, *but*, and *let me check*. Each event must have an entropy window from 15 tokens before through 50 tokens after the trigger. We inspect the next 360 characters for retraction cues such as *wrong*, *I made a mistake*, or *recompute*, and for agreement cues such as *yes*, *correct*, or *checks out*. An event is retained only when exactly one cue group matches. Multiple eligible forks from the same response are retained.

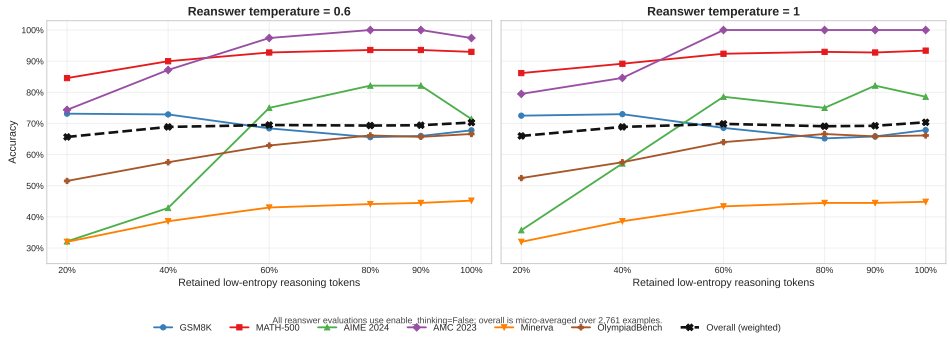


Figure S5: Qwen3-8B no-thinking re-answer accuracy after retaining different fractions of low-entropy reasoning tokens, evaluated at re-answer temperatures 0.6 and 1.0. Benchmark curves respond differently to retention, while the weighted aggregate is relatively stable. The variation cautions against treating low entropy as a universal deletion criterion.

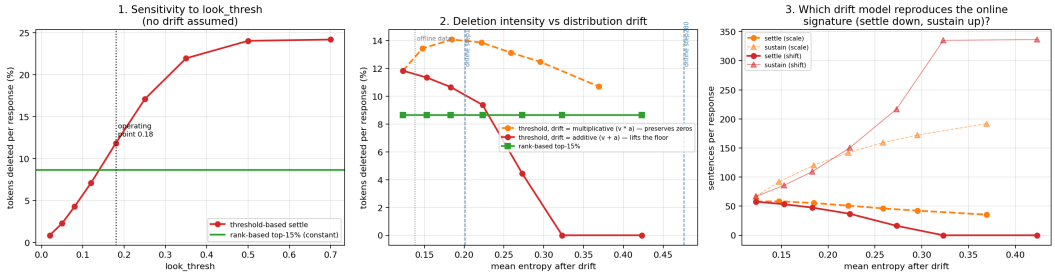


Figure S6: Offline sensitivity and distribution-drift diagnostics. Left: deletion intensity as the look threshold varies, with the analyzed operating point marked. Middle: deletion dose under multiplicative, additive, and rank-based perturbations. Right: synthetic drift signatures under scale and shift models. These plots characterize calibration sensitivity; they are not additional reinforcement-learning runs.

Labels are fixed before entropy is examined. The event baseline is the mean entropy at relative token positions -15 through -5, and the post-fork value is the mean at positions 10 through 30 minus that baseline. We compare the event-mean post-fork values and compute percentile 95% intervals from 2,000 bootstrap resamples of responses, keeping all forks from each sampled response together.

The full-sample analysis identifies 932 correction and 4,743 confirmation forks among 1,319 GSM8K responses, and 3,402 correction and 4,874 confirmation forks among 500 MATH responses. Post-fork entropy above baseline is higher for correction by 0.0252 nats on GSM8K with a 95% confidence interval of [0.0151, 0.0359], and by 0.0326 nats on MATH with an interval of [0.0247, 0.0406]. These estimates are calculated directly from the full-sample rollouts rather than digitized from Figure 1. The labels remain lexical proxies because cue matching does not compare the mathematical content of the earlier and later conclusions.

F.2 Correctness-Conditioned Length Measurement

We first audit whether mean output length is a reliable efficiency measure under no-thinking decoding. Figure S3 reports accuracy and response length across three Qwen models, two sampling strategies, and five mathematical reasoning benchmarks. The gap between all-response and correct-only length is substantial on several benchmarks, especially when truncation is frequent. Thus, an apparently long model can simply be spending more tokens on failed or truncated trajectories rather than performing more useful reasoning.

Figure S4 presents the same audit on a joint accuracy–length surface. Conditioning length on correctness moves many cells leftward, but the benchmark-dependent structure remains: harder cells generally occupy the lower-accuracy region, and their length varies strongly with both model and sampling strategy. This observation motivates reporting accuracy and length together and keeping the aggregation scope explicit throughout the paper.

F.3 Limits of Entropy-Only Filtering

We then test the stronger heuristic that low-entropy reasoning is uniformly dispensable. Figure S5 reanswers problems after retaining different fractions of low-entropy reasoning tokens. The aggregate curve is comparatively stable, but individual benchmarks respond differently: some improve only when most low-entropy context is retained, whereas others remain stable over a broader range. Thus token entropy alone does not identify a task-independent deletion rule. Self-Tidy instead asks whether a localized uncertainty excursion is subsequently resolved and assigns credit at the sentence level.

F.4 Detector Sensitivity and Distribution Drift

Figure S6 records an offline proxy analysis of the detector's deletion dose. The selected operating point lies on a steep part of the threshold curve, and additive distribution shifts can change the deletion floor more sharply than rank-based selection. This diagnostic motivates the response-relative quantiles and explicit budget used by Self-Tidy. We treat it as a sensitivity and boundary analysis rather than as an additional trained-policy result.