
Relaxed On-Policy Distillation: Selective Credit Allocation for Scaling Reasoning Efficiently

Jongwoo Ko¹ Sara Abdali¹ Young Jin Kim¹ Tianyi Chen¹ Pashmina Cameron¹

Abstract

On-policy distillation (OPD) is a promising approach for transferring reasoning capabilities to capacity-constrained models, yet sampled-token OPD often suffers from entropy collapse and negative transfer. We identify uniform credit allocation as a key bottleneck: existing objectives obtain dense teacher feedback on student-generated trajectories, but apply it uniformly across sampled tokens despite highly heterogeneous token-level signals. Most tokens carry redundant near-zero credit, while rare heavy-tailed negative credits can dominate updates and prematurely suppress plausible reasoning trajectories. We introduce REOPOLD (**Relaxed On-Policy Distillation**), a framework that relaxes this uniform assignment by controlling where teacher feedback is applied, how strongly it affects the update, and when the allocation rule changes during training. Across diverse reasoning tasks, REOPOLD improves sampled-token OPD over recent post-training baselines with up to $12\times$ higher sample efficiency, and extends to cross-vocabulary distillation, self-distillation, and test-time scaling.

1. Introduction

Large language models (LLMs) have achieved remarkable reasoning capabilities through large-scale post-training and test-time scaling, as demonstrated by recent reasoning-oriented systems such as OpenAI’s o1/o3 (Jaech et al., 2024) and DeepSeek-R1 (Guo et al., 2025). However, replicating this success in small language models (SLMs) proves difficult. Due to limited capacity, SLMs often benefit less efficiently from direct reward optimization, making standard reinforcement learning (RL) less effective for improving reasoning in capacity-constrained models (Guo et al.,

2025; Dang & Ngo, 2025; Yan et al., 2025). This disparity necessitates alternative mechanisms specifically tailored for transferring reasoning abilities to capacity-constrained models.

To address this, recent work (Yang et al., 2025; Patiño et al., 2025) has adopted on-policy distillation (OPD), where the student learns from its own trajectories under the guidance of a high-capacity teacher with stronger reasoning capabilities. By providing dense teacher-derived feedback on student-sampled trajectories, OPD reduces the burden of reward design while preserving the on-policy nature of training. Empirically, OPD has been shown to outperform RL and supervised fine-tuning (SFT) on reasoning tasks, making it a promising framework for transferring reasoning capabilities. More recently, sampled-token supervision (Lu et al., 2025; Xiao et al., 2026) has emerged as a common alternative to full-vocabulary distillation (Agarwal et al., 2024; Ko et al., 2024), as it avoids scoring the entire vocabulary and therefore scales more naturally to long reasoning trajectories.

However, a central bottleneck in OPD lies not only in acquiring teacher feedback, but also in allocating it effectively. Existing sampled-token methods compute teacher-derived credits (*i.e.*, log-likelihood ratios) on student-generated trajectories, but largely apply these credits uniformly over all sampled tokens. This uniform allocation overlooks the highly imbalanced nature of token-level credits: most sampled tokens provide redundant near-zero signals, whereas rare but heavy-tailed negative credits can dominate optimization, as shown in Figure 1. As a result, stable and efficient OPD requires mechanisms that decide *where* teacher feedback should be applied, *how much* it should influence the update, and *when* it should be trusted.

Empirically, this imbalance manifests as instability in standard OPD (Gudibande et al., 2023; Gu et al., 2024). We observe frequent negative transfer, where the student degrades relative to its base initialization (Section F and Figure 9), along with rapid entropy collapse that leads to premature convergence (Section 2 and Figure 3). This motivates our selective credit allocation view, which is related to prior work on non-uniform sample allocation and robust example reweighting (Schaul et al., 2015; Shrivastava et al., 2016;

¹Microsoft. Correspondence to: Jongwoo Ko <jongwooko@microsoft.com>.

Lin et al., 2017; Ren et al., 2018). Rather than treating all sampled-token credits uniformly, OPD should selectively allocate teacher supervision according to the student’s current learning dynamics.

Contributions. We analyze sampled-token OPD as a problem of selective credit allocation. Instead of treating all teacher-derived token signals as equally useful, we ask where teacher feedback should be applied, how strongly it should influence the update, and when the allocation rule should change during training. Our contributions are summarized as follows:

- **Credit allocation diagnosis:** We show that standard OPD implicitly applies a uniform credit allocation rule over sampled tokens. This rule is poorly matched to the heterogeneous credit structure of OPD, where a large fraction of sampled tokens carry near-zero credit, while rare teacher-student mismatches induce heavy-tailed negative credit and premature entropy collapse.
- **Student-conditioned credit allocation:** To address these, we propose **REOPOLD**, a unified framework that selectively regulates teacher credit along three axes: update magnitude, token uncertainty, and training stage. Concretely, REOPOLD integrates bounded negative credit, entropy-guided token selection, and exploration-to-refinement scheduling to stabilize sampled-token distillation.
- **State-of-the-art efficiency and scalability:** Empirically, REOPOLD achieves superior training sample efficiency and enables stronger test-time scaling for SLMs (Figure 6). Across mathematical, visual (Liu et al., 2025b), and tool-use (Qian et al., 2025) reasoning tasks, REOPOLD consistently outperforms baselines and remains effective beyond same-family distillation, including cross-vocabulary (Patiño et al., 2025; Boizard et al., 2025) and privileged self-distillation (Zhao et al., 2026).

2. Motivation: Indiscriminate Credit Allocation in On-Policy Distillation

2.1. Background: Sampled-token on-policy distillation

Let π_θ denote the student policy and π_T the teacher policy. For a query q , OPD samples responses from the student’s behavior policy $\pi_{\theta_{\text{old}}}$ and updates the student using teacher feedback on the sampled tokens. A common sampled-token reverse-KL objective, written in maximization form, is

$$\mathcal{J}_{\text{RKL}}(\theta) = \mathbb{E}_{q \sim \mathcal{Q}, \{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{\sum_i |o_i|} \sum_{i=1}^G \sum_{t=1}^{|o_i|} \rho_{i,t}(\theta) R_{i,t}(\theta) \right], \quad (1)$$

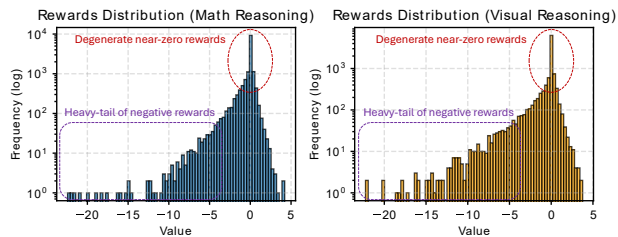


Figure 1. Log-scale histogram of teacher-derived credit. The red dashed ellipse indicates degenerate near-zero credits, while the purple dashed region highlights the heavy tail of negative credits.

where $\rho_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q, o_{i,<t})}$ and $R_{i,t}(\theta) = \log \frac{\pi_T(o_{i,t}|q, o_{i,<t})}{\pi_\theta(o_{i,t}|q, o_{i,<t})}$. We refer to $R_{i,t}$ as the teacher-derived token credit. Following sampled-token OPD practice, we detach $R_{i,t}$ and treat it as a fixed token-level credit within each update; the corresponding gradient derivation is provided in Appendix B.1.

This sampled-token formulation differs from full-vocabulary distillation (Agarwal et al., 2024; Gu et al., 2024; Ko et al., 2024), which compares teacher and student distributions over the entire vocabulary at each generated position. While full-vocabulary supervision provides richer feedback, it is memory-intensive for modern reasoning distillation with long trajectories and multiple rollouts (Table 10). A lightweight alternative is to evaluate the teacher only on the token sampled by the student, which has become widely used in recent OPD work (Lu et al., 2025; Xiao et al., 2026; Yang et al., 2026). We also provide detailed related work in Appendix A and empirical comparisons with full-vocabulary OPD and their corresponding Top- K approximations in Table 10.

2.2. Diagnosing Credit Misallocation in Sampled-Token OPD

Sampled-token OPD provides teacher feedback only on tokens sampled by the student, but standard objectives (Eq. 1) aggregate these credits uniformly across samples. This treats all sampled tokens as equally useful supervision targets, despite many credits being uninformative, redundant, or extreme. Because token-level credits are highly skewed in practice, this uniform allocation creates two finite-sample failure modes.

Most token credits are redundant. As shown in Figure 1, the sampled-token credit distribution has a large mass near zero. These credits occur when the teacher and student assign similar probability to the sampled token, yielding little corrective signal. While such tokens are not harmful individually, uniform aggregation makes them consume normalization mass and mini-batch budget that could be allocated to more informative tokens. As a result, many updates are diluted by nearly inactive credits, reducing the effective learning signal obtained from each sampled batch.

Table 1. Update-energy (Eq. 2) in negative-credit tails. Table 2. Retention (Eq. 3) on base samples during training.

Bottom	Math	Visual	Step (k)	Math	Visual
$\sim 1\%$	95.9%	97.8%	0	-1.583	-1.427
$\sim 5\%$	97.1%	98.6%	10	-2.554	-2.315
$\sim 10\%$	98.4%	99.5%	300	-2.841	-2.741

A few negative credits dominate the update. The more severe failure mode arises from the negative tail of the credit distribution. As shown in Figure 1, when the student samples a token that the teacher assigns very low probability, the credit $R_{i,t}(\theta)$ can become highly negative. Although this tail accounts for only a small fraction of sampled positions, its large magnitude can dominate optimization. To measure this concentration, we compute a logit-level update-energy proxy,

$$E_{i,t}(\theta) = R_{i,t}(\theta)^2 (1 - \|\pi_\theta(\cdot | q, o_{i,<t})\|_2^2), \quad (2)$$

which serves as an expected logit-level proxy for the squared gradient and allows us to decompose update energy by credit quantile. This proxy assumes stop-gradient through $R_{i,t}$, replacing the token-specific score norm by its expectation $1 - \|\pi_\theta\|_2^2$; see Appendix B.2 for the derivation. We find that the bottom 1% most negative credits account for over 95% of the total update energy in early standard OPD training (Table 1).

This concentration is harmful because negative credit directly suppresses the probability of sampled tokens (Wu et al., 2025; Ko et al., 2025a). In autoregressive models, suppressing an intermediate token can also disrupt the subsequent reasoning path conditioned on it. Thus, standard OPD may reduce support for trajectories that are locally disfavored by the teacher but still useful for exploration. We therefore use a base-policy retention diagnostic to measure how much support later checkpoints assign to trajectories sampled from the initial student π_{θ_0} :

$$\text{Ret}(k) = \mathbb{E}_{q, o \sim \pi_{\theta_0}} \left[\frac{1}{|o|} \sum_t \log \pi_{\theta_k}(o_t | q, o_{<t}) \right]. \quad (3)$$

Lower values mean that the current policy assigns less probability to trajectories originally generated by the base policy. Standard OPD rapidly reduces this support: within the first 10 steps, retention decreases from -1.58 and -1.43 to -2.55 and -2.32 (Table 2), corresponding to about 40% of the initial token-level likelihood. Together with the update-energy concentration and entropy collapse in Figure 3, this suggests rapid policy concentration under standard OPD.

Useful teacher signal is student-conditioned. These observations show that teacher feedback is not uniformly useful across sampled tokens. Near-zero credits provide little

corrective signal, while rare large negative credits may capture real teacher-student disagreement but destabilize optimization at full strength. The bottleneck in standard OPD is therefore credit allocation: which sampled-token credits to use, how strongly to apply them, and when to change the allocation rule during training.

3. REOPOLD: Relaxed On-Policy Distillation via Selective Credit Allocation

The analysis in Section 2 motivates a selective credit allocation view of OPD. Under this view, an objective should decide *where* teacher feedback should be applied, *how much* credit should be allowed to influence the update, and *when* negative correction should be trusted. REOPOLD instantiates these choices through **entropy-guided token selection**, **bounded negative credit**, and an **exploration-to-refinement schedule**. For each batch \mathcal{B} sampled from the query set \mathcal{Q} , we maximize:

$$\mathcal{J}_{\text{REOPOLD}}(\theta) = \mathbb{E}_{\mathcal{B} \sim \mathcal{Q}, q \sim \mathcal{B}, \{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(\cdot | q)} \left[\frac{1}{\sum_{i=1}^G \sum_{t=1}^{|o_i|} M_{i,t}^{(k)}} \sum_{i=1}^G \sum_{t=1}^{|o_i|} \rho_{i,t}(\theta) \hat{R}_{i,t}^\lambda(\theta) M_{i,t}^{(k)} \right], \quad (4)$$

where

$$\hat{R}_{i,t}^\lambda(\theta) = \max \left(\text{sg}(R_{i,t}(\theta)), \frac{\log \lambda}{1 - \lambda} \right), \quad (5)$$

$$M_{i,t}^{(k)} = \begin{cases} \mathbb{I} \left[R_{i,t}(\theta) \geq \frac{\log \lambda}{1 - \lambda} \right] & \text{if } k < T_{\text{switch}} \\ \mathbb{I} \left[H_t^i \geq \tau_\beta \right] & \text{if } k \geq T_{\text{switch}} \end{cases}, \quad (6)$$

The full training procedure is summarized in Algorithm 1.

3.1. Where to Allocate Credit: Entropy-Guided Token Selection

The first axis of credit allocation concerns *where* teacher feedback should be applied. Standard OPD allocates update budget uniformly across all sampled tokens, even though many near-zero credit tokens occur at low-entropy positions. As shown in Figure 2, these tokens are concentrated in contexts where the student and teacher assign similar probabilities to the sampled token, making additional correction largely redundant. The following elementary softmax identity supports using entropy as a lightweight allocation signal.

Remark 3.1. Let $z \in \mathbb{R}^{|\mathcal{V}|}$ be the logit vector over a finite vocabulary \mathcal{V} , and let $p = \text{softmax}(z)$ be the categorical distribution, with p_v denoting the probability of token $v \in \mathcal{V}$. If $y \sim p$, then

$$\mathbb{E}_{y \sim p} [\|\nabla_z \log p_y\|_2^2] = 1 - \|p\|_2^2 \leq H(p), \quad (7)$$

where p_y is the probability of the sampled token and $H(p) = -\sum_{v \in \mathcal{V}} p_v \log p_v$.

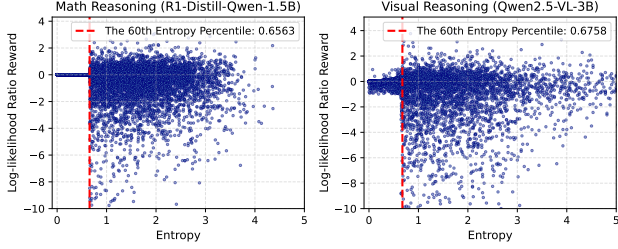


Figure 2. **Correlation between token entropy and credit.** Experimental results on math reasoning and visual reasoning benchmarks demonstrate that credits in the bottom 60th entropy percentile are heavily concentrated around zero. This suggests that while teacher and student policies may diverge overall, they remain highly consistent on low-entropy, deterministic tokens, with significant deviations occurring primarily in high-entropy regimes.

Applied to $p = \pi_\theta(\cdot|x)$ at context $x = (q, o_{<t})$, $\nabla_z \log p_y$ is the logit-level score of the sampled token, which captures how much a token-level likelihood update can locally redistribute probability mass (Williams, 1992; Kakade, 2001). Remark 3.1 shows that entropy upper-bounds the expected squared score norm, which captures local update leverage in logit space; the derivation is given in Appendix B.3. The teacher credit provides the direction and magnitude of supervision, while student entropy indicates whether the student distribution at that position remains plastic enough for this signal to induce a meaningful redistribution.

Motivated by this, we use student entropy as a lightweight allocation signal. For each sampled token $o_{i,t}$, let H_t^i denote the entropy of the student’s policy $\pi_\theta(\cdot|q, o_{i,<t})$. We define a token-level allocation mask

$$M_{i,t}^{\text{ent}} = \mathbb{I} [H_t^i \geq \tau_\beta], \quad (8)$$

where τ_β is chosen so that $M_{i,t}^{\text{ent}}$ selects the top- β fraction of token entropies within the batch. This mask selects positions where the student remains uncertain, rather than uniformly accepting teacher credit at every sampled token. Integrating this mask into the sampled-token objective gives

$$\mathcal{J}_{\text{Ent}}(\theta) = \mathbb{E}_{\mathcal{B} \sim \mathcal{Q}, q \sim \mathcal{B}, \{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{\sum_{i=1}^G \sum_{t=1}^{|o_i|} \rho_{i,t}(\theta) R_{i,t}(\theta) M_{i,t}^{\text{ent}}}{\sum_{i=1}^G \sum_{t=1}^{|o_i|} M_{i,t}^{\text{ent}}} \right]. \quad (9)$$

This creates a dynamic token batch whose effective size is determined by the student’s uncertainty rather than sequence length. By filtering out low-entropy, near-zero-credit tokens, entropy-guided allocation reduces gradient dilution and concentrates update budget on positions where teacher feedback is most likely to be useful. Empirically, the sensitivity analysis in Figure 4 confirms that stricter entropy selection accelerates convergence and improves final accuracy, supporting the view that selective allocation of teacher credit is more effective than uniform token-level imitation.

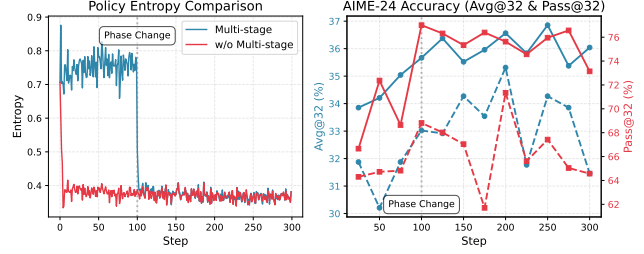


Figure 3. **Analysis of exploration-to-refinement scheduling.** (Left) The proposed multi-stage approach prevents entropy collapse and preserves diverse modes during the initial exploration phase by masking strongly negative credits, whereas the baseline suffers from rapid entropy collapse. (Right) The multi-stage strategy (solid lines) achieves superior final performance to the baseline (dashed lines) in both quality (i.e., Avg@32) and diversity (i.e., Pass@32).

3.2. How Much Credit to Use: Bounded Negative Credit

The second allocation axis controls how strongly the student should absorb negative teacher feedback. If the student samples a token with negligible teacher probability, the log-ratio credit can become extremely negative and dominate the update. We bound this tail using a mixture relaxation of the teacher and student distributions. For $\lambda \in [0, 1)$, Jensen’s inequality gives

$$R_{i,t}(\theta) = \log \frac{\pi_T}{\pi_\theta} \leq \frac{1}{1-\lambda} \log \frac{(1-\lambda) \cdot \pi_T + \lambda \pi_\theta}{\pi_\theta}.$$

The key insight comes from the asymptotic behavior of the relaxed term; detailed derivation is provided in Appendix B.4. While the original credit $R_{i,t}(\theta)$ diverges to $-\infty$ as $\pi_T(o_{i,t}|q, o_{i,<t}) \rightarrow 0$, the relaxed expression converges to $c_\lambda = \frac{\log \lambda}{1-\lambda}$. We use this limiting value as a stable finite penalty motivated by the mixture-relaxed bound, and apply it as a clipping floor for heavy-tailed negative credits:

$$\hat{R}_{i,t}^\lambda(\theta) = \max(R_{i,t}(\theta), c_\lambda). \quad (10)$$

In the credit allocation view, this operation decides how much a single negative teacher signal is allowed to influence the student. It preserves the objective for ordinary tokens, while preventing rare teacher-student mismatches from consuming disproportionate update budget. Unlike Skew RKL (Ko et al., 2024), which changes the full objective and is sensitive to λ , this rule only targets outlying negative credits, yielding the robustness shown in Figure 5.

3.3. When to Change Allocation: Exploration-to-Refinement Scheduling

The third allocation axis controls *when* different teacher credits should be accepted. The usefulness of teacher feedback changes over training: early updates should preserve plausible student trajectories, while later updates should

Table 3. Performance comparison of vision-language models on visual reasoning and perception benchmarks. Accuracy (%) is reported for all benchmarks. The best result in each column is shown in **bold**, and the second-best is underlined. † denotes results obtained with the Top-5 approximation, as the full-vocabulary implementation incurs memory issues. Our reproduced OPD results are reported as mean ± standard deviation over 5 runs.

Model	Geo3K	MathVerse	MathVision	MathVista	WeMath	Hallusion	AVG.
Qwen2.5-VL-32B-Instruct + NoisyRollout	56.74	58.86	39.82	78.30	75.51	72.45	63.61
Qwen2.5-VL-32B-Instruct + NoisyRollout (Liu et al., 2025b) → Qwen2.5-VL-3B-Instruct							
Qwen2.5-VL-3B-Instruct	26.46±1.58	35.58±0.44	22.83±0.66	59.40±0.49	53.41±0.41	61.51±0.39	43.20±0.54
+ PAPO (Wang et al., 2025)	32.95	<u>40.65</u>	24.16	65.10	63.62	<u>61.62</u>	<u>48.02</u>
+ Standard OPD (Lu et al., 2025)	<u>48.09</u> ±0.39	33.25±0.41	<u>24.78</u> ±0.62	<u>62.50</u> ±0.56	<u>64.48</u> ±0.43	60.83±0.38	48.99±0.34
+ REOPOLD	50.58 ±0.64	46.40 ±0.40	26.39 ±0.38	61.50±0.45	64.60 ±0.23	63.62 ±0.21	52.18 ±0.21
Qwen2.5-VL-32B-Instruct + NoisyRollout (Liu et al., 2025b) → Qwen2.5-VL-7B-Instruct							
Qwen2.5-VL-7B-Instruct	39.77±1.33	45.72±0.69	25.05±0.68	67.80±0.45	64.77±0.53	65.62±1.39	51.46±0.13
+ GRPO (Shao et al., 2024)	51.4	50.8	27.3	70.5	67.4	69.8	56.20
+ NoisyRollout (Liu et al., 2025b)	50.08	53.14	26.64	<u>72.00</u>	70.57	70.66	<u>57.18</u>
+ Standard OPD (Lu et al., 2025)	<u>51.75</u> ±0.40	47.71±0.61	<u>28.79</u> ±0.32	<u>71.27</u> ±0.68	<u>70.06</u> ±0.64	69.51±1.29	56.51±0.19
+ GKD† (Agarwal et al., 2024)	52.07±0.84	48.31±0.47	27.98±0.53	70.89±0.36	69.34±0.41	69.75±1.13	56.39±0.24
+ REOPOLD	53.58 ±0.71	<u>51.43</u> ±0.46	29.21 ±0.49	72.40 ±0.68	69.77±0.24	<u>70.14</u> ±0.86	57.76 ±0.31

refine unresolved token decisions. We implement this temporal control through a token-wise allocation mask:

$$\mathcal{J}_{\text{REOPOLD}}(\theta) = \mathbb{E}_{\mathcal{B} \sim \mathcal{Q}, q \sim \mathcal{B}, \{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{\sum_{i=1}^G \sum_{t=1}^{|o_i|} \rho_{i,t}(\theta) \hat{R}_{i,t}^\lambda(\theta) M_{i,t}^{(k)}}{\sum_{i=1}^G \sum_{t=1}^{|o_i|} M_{i,t}^{(k)}} \right]. \quad (11)$$

Here, $M_{i,t}^{(k)}$ determines which token credits are accepted at training step k . By changing this mask over time, REOPOLD uses different allocation rules for exploration and refinement.

Exploration phase. The goal of the early phase is to preserve exploration while avoiding destructive teacher corrections. For $k < T_{\text{switch}}$, REOPOLD keeps dense supervision but filters out excessively negative credits:

$$M_{i,t}^{(k)} = \mathbb{I} \left[R_{i,t}(\theta) \geq \frac{\log \lambda}{1 - \lambda} \right]. \quad (12)$$

This mask rejects only credits whose negative magnitude exceeds the bounded-credit threshold in Section 3.2, delaying overly aggressive correction that could prematurely prune candidate reasoning paths. As shown in Figure 3, the first phase improves both Avg@32 and Pass@32, suggesting that bounded negative correction preserves useful exploration rather than merely slowing optimization.

Refinement phase. The goal of the later phase is to concentrate update budget on unresolved decisions. For $k \geq T_{\text{switch}}$, REOPOLD switches to the entropy-guided allocation rule from Section 3.1:

$$M_{i,t}^{(k)} = \mathbb{I} [H_t^i \geq \tau_\beta]. \quad (13)$$

This mask selects high-entropy positions where the student remains uncertain and teacher feedback is more actionable. Thus, the schedule first preserves exploratory support and

then refines uncertain token decisions. Empirically, Figure 3 shows that this transition further improves Avg@32 while maintaining the diversity gains reflected by Pass@32.

4. Main Results: Visual Reasoning

We present the primary evaluation results on visual reasoning tasks below. Please refer to Appendix E for additional ablation studies and detailed discussions. We further provide experimental results on mathematical reasoning with multiple teacher-student pairs including cross-vocabulary setups (Appendix F) and agentic reasoning with visual tool-use (Appendix G).

Setup. We adopt Qwen2.5-VL-3/7B-Instruct (Bai et al., 2025) as the student policy and Qwen2.5-VL-32B-Instruct, trained with NoisyRollout (Liu et al., 2025b), as the teacher. We train the student model on Geometry3K (Lu et al., 2021), which focuses on geometric problem solving and comprises approximately 2.1K training samples. Following the protocol of Liu et al. (2025b), we pre-process this dataset by converting all multiple-choice questions into free-form answer formats to mitigate reward hacking and reduce the likelihood of answer guessing. Detailed training setup is described in Appendix D.

Evaluation. We assess across six benchmarks: five visual reasoning benchmarks, including test split of Geometry3K, MathVerse (Zhang et al., 2024a), MathVision (Wang et al., 2024), MathVista (Lu et al., 2023), and WeMath (Qiao et al., 2025), as well as one visual perception benchmark, HallusionBench (Guan et al., 2024). By following the evaluation protocol of Liu et al. (2025b), we employ greedy sampling and nucleus sampling (Holtzman et al., 2020) with a temperature of 0.6 and a top-p of 0.95 for model inference and use Gemini-2.0-Flash-001 (Team et al., 2023) as the judge model to parse generated responses.

Table 4. Performance comparison of Qwen2.5-VL-3/7B-Instruct trained with 32B teacher models on visual reasoning benchmarks. Accuracy (%) is reported for all benchmarks.

	(1)	(2)	(3)	Geo3K	Verse	Vision	AVG.
Qwen2.5-VL-3B-Instruct							
Standard OPD				48.09	33.25	24.78	35.37
	✓			48.59	45.53	25.07	39.73
	✓	✓		50.08	45.94	26.42	40.81
REOPOLD	✓	✓	✓	50.58	46.40	26.39	41.12
Qwen2.5-VL-7B-Instruct							
Standard OPD				51.75	47.71	28.79	42.75
	✓			51.75	51.45	29.05	44.08
	✓	✓		52.75	51.42	29.10	44.42
REOPOLD	✓	✓	✓	53.58	51.43	29.21	44.74

Result 1: Efficacy on compact models. As shown in Table 3, REOPOLD achieves superior overall performance compared to standard OPD, GKD (Agarwal et al., 2024), and RL baselines across visual reasoning and perception benchmarks for both 3B and 7B models. REOPOLD also surpasses specialized perception algorithms like NoisyRollout (7B) and PAPO (Wang et al., 2025; 3B). The Top-*K* study in Appendix E further confirms that REOPOLD extends beyond sampled-token supervision by outperforming Top-5 approximated full-vocabulary OPD methods. This highlights its robustness for compact models: by selectively applying teacher credits, REOPOLD avoids unnecessary imitation that can be harmful in low-capacity regimes.

Result 2: Impact of module design. Table 4 validates the three axes of selective credit allocation: (1) bounded negative credit (2) entropy-guided token selection, and (3) exploration-to-refinement scheduling. Each component improves over standard OPD by addressing a distinct allocation failure: overly large negative credits, redundant low-entropy tokens, and the changing usefulness of teacher feedback over training. Combining all three achieves the best average performance for both 3B and 7B students. Consistently, Figure 4 shows that stricter high-entropy token selection (e.g., $\beta = 0.2$ or $\beta = 0.5$) accelerates convergence and improves final accuracy, supporting the view that filtering redundant low-entropy tokens mitigates gradient dilution. We refer the reader to Figure 5 and Figure 3 for extended analyses on bounded negative credit and exploration-to-refinement scheduling.

Result 3: Extension to self-distillation. We further test whether selective credit allocation extends to on-policy self-distillation (OPSD). In OPSD, the same model serves as both teacher and student: the teacher policy is conditioned on privileged information, such as ground-truth solutions, while the student policy observes only the original prompt (Zhao et al., 2026; Shenfeld et al., 2026). Apply-

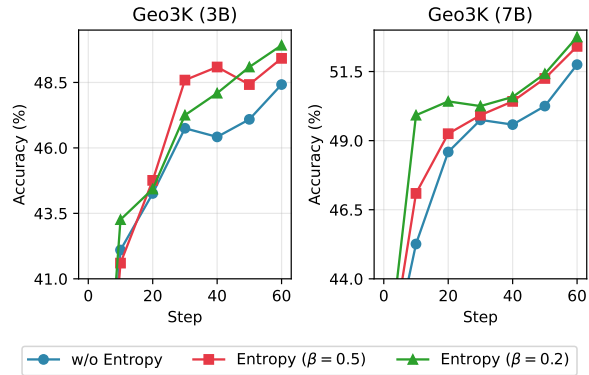


Figure 4. Comparison among REOPOLD using different entropy-selection fractions. Our token selection accelerates convergence and leads to superior final accuracy.

Table 5. Performance comparison of Qwen2.5-VL-3/7B-Instruct trained by OPSD (Zhao et al., 2026) on visual reasoning benchmarks. Accuracy (%) is reported for all benchmarks.

	Geo3K	Verse	Vision	Vista	AVG.
Qwen2.5-VL-3B-Instruct					
OPSD	44.27	41.94	24.16	61.75	43.03
Relaxed OPSD	49.82	45.14	25.71	61.82	45.62
Qwen2.5-VL-7B-Instruct					
OPSD	50.72	49.41	28.22	70.33	49.67
Relaxed OPSD	51.82	50.43	29.07	70.84	50.54

ing our selective credit allocation strategy to OPSD yields Relaxed OPSD. As shown in Table 5, Relaxed OPSD consistently improves average accuracy over OPSD for both Qwen2.5-VL-3B-Instruct (43.03 to 45.62) and Qwen2.5-VL-7B-Instruct (49.67 to 50.54). This shows that selective credit allocation is not limited to external-teacher distillation, but also stabilizes privileged self-distillation.

5. Conclusion

We studied sampled-token OPD through the lens of selective credit allocation. Standard OPD applies teacher credits uniformly despite their imbalanced structure, causing inefficient updates and premature suppression of useful reasoning paths. To address this, we introduced REOPOLD, a unified framework that regulates teacher feedback along three axes: entropy-guided token selection focuses supervision on uncertain decisions, bounded negative credit controls update magnitude, and an exploration-to-refinement schedule changes the allocation rule over training. Empirically, REOPOLD improves OPD across mathematical, visual, and agentic tool-use reasoning, and remains effective under cross-vocabulary distillation, self-distillation, longer training, and test-time scaling. These results suggest selective credit allocation is a practical principle for robust reasoning distillation in compact models.

References

- Acharya, M., Kafle, K., and Kanan, C. Tallyqa: Answering complex counting questions. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 8076–8084, 2019.
- Agarwal, R., Vieillard, N., Zhou, Y., Stanczyk, P., Garea, S. R., Geist, M., and Bachem, O. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=3zKtaqxLhW>.
- Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.
- Barbiero, P., Ciravegna, G., Giannini, F., Zarlenga, M. E., Magister, L. C., Tonda, A., Lio, P., Precioso, F., Jamnik, M., and Marra, G. Interpretable neural-symbolic concept reasoning. In *ICML 2023 Workshop on Differentiable Almost Everything: Differentiable Relaxations, Algorithms, Operators, and Simulators*, 2023. URL <https://openreview.net/forum?id=oRj82I2apn>.
- Boizard, N., Haddad, K. E., HUDELLOT, C., and Colombo, P. Towards cross-tokenizer distillation: the universal logit distillation loss for LLMs. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL <https://openreview.net/forum?id=bwRxXiG09A>.
- Calanzone, D., Teso, S., and Vergari, A. Logically consistent language models via neuro-symbolic integration. In *The Thirteenth International Conference on Learning Representations*, 2025. URL <https://openreview.net/forum?id=7Pgluppo4k>.
- Chen, A., Li, A., Gong, B., Jiang, B., Fei, B., Yang, B., Shan, B., Yu, C., Wang, C., Zhu, C., et al. Minimax-ml: Scaling test-time compute efficiently with lightning attention. *arXiv preprint arXiv:2506.13585*, 2025a.
- Chen, Q., Qin, L., Liu, J., Peng, D., Guan, J., Wang, P., Hu, M., Zhou, Y., Gao, T., and Che, W. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models. *arXiv preprint arXiv:2503.09567*, 2025b.
- Creswell, A., Shanahan, M., and Higgins, I. Selection-inference: Exploiting large language models for interpretable logical reasoning. *arXiv preprint arXiv:2205.09712*, 2022.
- Cui, G., Zhang, Y., Chen, J., Yuan, L., Wang, Z., Zuo, Y., Li, H., Fan, Y., Chen, H., Chen, W., et al. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.
- Dang, Q.-A. and Ngo, C. Reinforcement learning for reasoning in small llms: What works and what doesn't. *arXiv preprint arXiv:2503.16219*, 2025.
- Giadikiaroglou, P., Lymperaiou, M., Filandrianos, G., and Stamou, G. Puzzle solving using reasoning of large language models: A survey. In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N. (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 11574–11591, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.646. URL <https://aclanthology.org/2024.emnlp-main.646/>.
- Gu, Y., Dong, L., Wei, F., and Huang, M. MiniLLM: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=5h0qf7IBZZ>.
- Guan, T., Liu, F., Wu, X., Xian, R., Li, Z., Liu, X., Wang, X., Chen, L., Huang, F., Yacoob, Y., et al. Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14375–14385, 2024.
- Gudibande, A., Wallace, E., Snell, C., Geng, X., Liu, H., Abbeel, P., Levine, S., and Song, D. The false promise of imitating proprietary llms. *arXiv preprint arXiv:2305.15717*, 2023.
- Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- Hao, S., Gu, Y., Ma, H., Hong, J., Wang, Z., Wang, D., and Hu, Z. Reasoning with language model is planning with world model. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 8154–8173, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.507. URL <https://aclanthology.org/2023.emnlp-main.507/>.
- He, C., Luo, R., Bai, Y., Hu, S., Thai, Z., Shen, J., Hu, J., Han, X., Huang, Y., Zhang, Y., Liu, J., Qi, L., Liu, Z., and Sun, M. OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In Ku, L.-W., Martins, A., and Sriku-mar, V. (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3828–3850, Bangkok, Thailand,

- August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.211. URL <https://aclanthology.org/2024.acl-long.211/>.
- He, J., Liu, J., Liu, C. Y., Yan, R., Wang, C., Cheng, P., Zhang, X., Zhang, F., Xu, J., Shen, W., et al. Skywork open reasoner 1 technical report. *arXiv preprint arXiv:2505.22312*, 2025a.
- He, Z., Liang, T., Xu, J., Liu, Q., Chen, X., Wang, Y., Song, L., Yu, D., Liang, Z., Wang, W., et al. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. *arXiv preprint arXiv:2504.11456*, 2025b.
- Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.
- Ho, N., Schmid, L., and Yun, S.-Y. Large language models are reasoning teachers. In Rogers, A., Boyd-Graber, J., and Okazaki, N. (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14852–14882, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.830. URL <https://aclanthology.org/2023.acl-long.830/>.
- Holtzman, A., Buys, J., Du, L., Forbes, M., and Choi, Y. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020. URL <https://openreview.net/forum?id=rygGQyrFvH>.
- Hsieh, C.-Y., Li, C.-L., Yeh, C.-k., Nakhost, H., Fujii, Y., Ratner, A., Krishna, R., Lee, C.-Y., and Pfister, T. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In Rogers, A., Boyd-Graber, J., and Okazaki, N. (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 8003–8017, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.507. URL <https://aclanthology.org/2023.findings-acl.507/>.
- Huang, J. and Chang, K. C.-C. Towards reasoning in large language models: A survey. In Rogers, A., Boyd-Graber, J., and Okazaki, N. (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 1049–1065, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.67. URL <https://aclanthology.org/2023.findings-acl.67/>.
- Jaech, A., Kalai, A., Lerer, A., Richardson, A., El-Kishky, A., Low, A., Helyar, A., Madry, A., Beutel, A., Carney, A., et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.
- Jiang, D., Lu, Y., Li, Z., Lyu, Z., Nie, P., Wang, H., Su, A., Chen, H., Zou, K., Du, C., et al. Verltool: Towards holistic agentic reinforcement learning with tool use. *arXiv preprint arXiv:2509.01055*, 2025.
- Kakade, S. M. A natural policy gradient. *Advances in neural information processing systems*, 14, 2001.
- Ko, J., Kim, S., Chen, T., and Yun, S.-Y. DistiLLM: Towards streamlined distillation for large language models. In *Forty-first International Conference on Machine Learning*, 2024. URL <https://openreview.net/forum?id=lsHZNNoc7r>.
- Ko, J., Chen, T., Kim, S., Ding, T., Liang, L., Zharkov, I., and Yun, S.-Y. DistiLLM-2: A contrastive approach boosts the distillation of LLMs. In *Forty-second International Conference on Machine Learning*, 2025a. URL <https://openreview.net/forum?id=rc65N9xIrY>.
- Ko, J., Kim, S., Cho, S., and Yun, S.-Y. Flex-judge: Text-only reasoning unleashes zero-shot multimodal evaluators. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b. URL <https://openreview.net/forum?id=v6kyF3S7dM>.
- Lewkowycz, A., Andreassen, A. J., Dohan, D., Dyer, E., Michalewski, H., Ramasesh, V. V., Slone, A., Anil, C., Schlag, I., Gutman-Solo, T., Wu, Y., Neyshabur, B., Gur-Ari, G., and Misra, V. Solving quantitative reasoning problems with language models. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022. URL <https://openreview.net/forum?id=IFXTZERXdm7>.
- Li, Q., Zhu, Y., Liang, Y., Wu, Y. N., Zhu, S.-C., and Huang, S. Neural-symbolic recursive machine for systematic generalization. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=FWJAmwE0xH>.
- Li, Y., Zuo, Y., He, B., Zhang, J., Xiao, C., Qian, C., Yu, T., Gao, H.-a., Yang, W., Liu, Z., et al. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026.
- Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

- Liu, M., Diao, S., Lu, X., Hu, J., Dong, X., Choi, Y., Kautz, J., and Dong, Y. Prorl: Prolonged reinforcement learning expands reasoning boundaries in large language models. *arXiv preprint arXiv:2505.24864*, 2025a.
- Liu, X., Ni, J., Wu, Z., Du, C., Dou, L., Wang, H., Pang, T., and Shieh, M. Q. Noisyrollout: Reinforcing visual reasoning with data augmentation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b. URL <https://openreview.net/forum?id=9zD2i7YRot>.
- Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL <https://openreview.net/forum?id=Bkg6RiCqY7>.
- Lu, K. et al. On-policy distillation. <https://thinkingmachines.ai/blog/on-policy-distillation/>, 2025. Thinking Machines Blog, accessed on 2025-10-27.
- Lu, P., Gong, R., Jiang, S., Qiu, L., Huang, S., Liang, X., and Zhu, S.-C. Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning. *arXiv preprint arXiv:2105.04165*, 2021.
- Lu, P., Bansal, H., Xia, T., Liu, J., Li, C., Hajishirzi, H., Cheng, H., Chang, K.-W., Galley, M., and Gao, J. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023.
- Luo, M., Tan, S., Wong, J., Shi, X., Tang, W. Y., Roongta, M., Cai, C., Luo, J., Zhang, T., Li, L. E., et al. Deepscaler: Surpassing o1-preview with a 1.5 b model by scaling rl. *Notion Blog*, 2025.
- Mathew, M., Bagal, V., Tito, R., Karatzas, D., Valveny, E., and Jawahar, C. Infographicvqa. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1697–1706, 2022.
- Patiño, C. M., Rasul, K., Gallouédec, Q., Burtenshaw, B., Paniago, S., Srivastav, V., Frere, T., Beeching, E., Tunstall, L., von Werra, L., and Wolf, T. Unlocking on-policy distillation for any model family. <https://huggingface.co/spaces/HuggingFaceH4/on-policy-distillation/>, 2025. HuggingFace Blog, accessed on 2025-10-29.
- Qian, C., Acikgoz, E. C., He, Q., Wang, H., Chen, X., Hakkani-Tür, D., Tur, G., and Ji, H. Toolrl: Reward is all tool learning needs. *arXiv preprint arXiv:2504.13958*, 2025.
- Qiao, R., Tan, Q., Dong, G., MinhuiWu, M., Sun, C., Song, X., Wang, J., Gongque, Z., Lei, S., Zhang, Y., et al. We-math: Does your large multimodal model achieve human-like mathematical reasoning? In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 20023–20070, 2025.
- Rajani, N. F., McCann, B., Xiong, C., and Socher, R. Explain yourself! leveraging language models for commonsense reasoning. In Korhonen, A., Traum, D., and Márquez, L. (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4932–4942, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1487. URL <https://aclanthology.org/P19-1487/>.
- Ren, M., Zeng, W., Yang, B., and Urtasun, R. Learning to reweight examples for robust deep learning. In *International conference on machine learning*, pp. 4334–4343. PMLR, 2018.
- Santana, M. A. B., Gallagher, K., Ielo, A., Kareem, I., Ricca, F., and Russo, A. Question answering with llms and learning from answer sets. *Theory and Practice of Logic Programming*, pp. 1–25, 2025.
- Schaul, T., Quan, J., Antonoglou, I., and Silver, D. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.
- Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.
- Shenfeld, I., Damani, M., Hübotter, J., and Agrawal, P. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*, 2026.
- Sheng, G., Zhang, C., Ye, Z., Wu, X., Zhang, W., Zhang, R., Peng, Y., Lin, H., and Wu, C. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 1279–1297, 2025.
- Shrivastava, A., Gupta, A., and Girshick, R. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 761–769, 2016.
- Su, A., Wang, H., Ren, W., Lin, F., and Chen, W. Pixel reasoner: Incentivizing pixel-space reasoning with curiosity-driven reinforcement learning. *arXiv preprint arXiv:2505.15966*, 2025.

- Team, G., Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.
- Wang, K., Pan, J., Shi, W., Lu, Z., Ren, H., Zhou, A., Zhan, M., and Li, H. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in Neural Information Processing Systems*, 37:95095–95169, 2024.
- Wang, X., Wei, J., Schuurmans, D., Le, Q. V., Chi, E. H., Narang, S., Chowdhery, A., and Zhou, D. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=1PLlNIMMrw>.
- Wang, Z., Guo, X., Stoica, S., Xu, H., Wang, H., Ha, H., Chen, X., Chen, Y., Yan, M., Huang, F., et al. Perception-aware policy optimization for multimodal reasoning. *arXiv preprint arXiv:2507.06448*, 2025.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.
- Weng, Y., Zhu, M., Xia, F., Li, B., He, S., Liu, K., and Zhao, J. Mastering symbolic operations: Augmenting language models with compiled neural networks. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=9nsNyN0vox>.
- Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.
- Wu, P. and Xie, S. V?: Guided visual search as a core mechanism in multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13084–13094, 2024.
- Wu, T., Tao, C., Wang, J., Yang, R., Zhao, Z., and Wong, N. Rethinking Kullback-Leibler divergence in knowledge distillation for large language models. In Rambow, O., Wanner, L., Apidianaki, M., Al-Khalifa, H., Eugenio, B. D., and Schockaert, S. (eds.), *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 5737–5755, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL <https://aclanthology.org/2025.coling-main.383/>.
- Xiao, B., Xia, B., Yang, B., Gao, B., Shen, B., Zhang, C., He, C., Lou, C., Luo, F., Wang, G., et al. Mimo-v2-flash technical report. *arXiv preprint arXiv:2601.02780*, 2026.
- Xu, H., Peng, B., Awadalla, H., Chen, D., Chen, Y.-C., Gao, M., Kim, Y. J., Li, Y., Ren, L., Shen, Y., et al. Phi-4-mini-reasoning: Exploring the limits of small reasoning language models in math. *arXiv preprint arXiv:2504.21233*, 2025a.
- Xu, H., Zhu, Q., Deng, H., Li, J., Hou, L., Wang, Y., Shang, L., Xu, R., and Mi, F. Kdrl: Post-training reasoning llms via unified knowledge distillation and reinforcement learning. *arXiv preprint arXiv:2506.02208*, 2025b.
- Yan, J., Li, Y., Hu, Z., Wang, Z., Cui, G., Qu, X., Cheng, Y., and Zhang, Y. Learning to reason under off-policy guidance. *arXiv preprint arXiv:2504.14945*, 2025.
- Yang, A., Zhang, B., Hui, B., Gao, B., Yu, B., Li, C., Liu, D., Tu, J., Zhou, J., Lin, J., et al. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024.
- Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 technical report. *arXiv preprint arXiv: 2505.09388*, 2025.
- Yang, W., Liu, W., Xie, R., Yang, K., Yang, S., and Lin, Y. Learning beyond teacher: Generalized on-policy distillation with reward extrapolation. *arXiv preprint arXiv:2602.12125*, 2026.
- Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023a.
- Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023b.
- Zhang, R., Jiang, D., Zhang, Y., Lin, H., Guo, Z., Qiu, P., Zhou, A., Lu, P., Chang, K.-W., Qiao, Y., et al. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In *European Conference on Computer Vision*, pp. 169–186. Springer, 2024a.
- Zhang, X., Du, C., Pang, T., Liu, Q., Gao, W., and Lin, M. Chain of preference optimization: Improving chain-of-thought reasoning in LLMs. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024b. URL <https://openreview.net/forum?id=2cczgOfMP4>.
- Zhao, S., Xie, Z., Liu, M., Huang, J., Pang, G., Chen, F., and Grover, A. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

Zhao, Y., Liu, Y., Liu, J., Chen, J., Wu, X., Hao, Y., Lv, T., Huang, S., Cui, L., Ye, Q., et al. Geometric-mean policy optimization. *arXiv preprint arXiv:2507.20673*, 2025.

Zhao, Z., Lee, W. S., and Hsu, D. Large language models as commonsense knowledge for large-scale task planning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=WjplAYB81H>.

Zheng, C., Liu, S., Li, M., Chen, X.-H., Yu, B., Gao, C., Dang, K., Liu, Y., Men, R., Yang, A., et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025.

A. Related Work

Reasoning models. Reasoning models represent a distinct class of machine learning systems designed to execute structured, logical, and multi-step inference over input queries (Creswell et al., 2022; Huang & Chang, 2023; Chen et al., 2025b). In contrast to standard predictive models that rely on direct input-to-output mapping, reasoning models emulate human cognitive processes by integrating learned knowledge with stepwise deduction (Creswell et al., 2022), chain-of-thought processing (Wei et al., 2022), or symbolic manipulation (Weng et al., 2024). These capabilities are essential for tasks involving complex problem-solving (Wang et al., 2023; Giadikiaroglou et al., 2024), question answering (Zhang et al., 2024b; Santana et al., 2025), planning (Yao et al., 2023a; Hao et al., 2023), and commonsense inference (Rajani et al., 2019; Zhao et al., 2023), where a single forward pass is often insufficient. Recent advancements have increasingly integrated reasoning into LLMs and neuro-symbolic architectures (Li et al., 2024; Calanzone et al., 2025). This integration allows models to decompose complex problems into intermediate steps, verify logical consistency (Calanzone et al., 2025), and generate interpretable solutions (Barbiero et al., 2023; Yao et al., 2023b). Fundamentally, reasoning models ensure that outputs are both accurate and justifiable, emphasizing the rationale behind a decision as much as the decision itself.

Policy optimization for reasoning models. Recent advancements in policy optimization focus on enhancing the sample efficiency, stability, and reasoning depth of LLMs. While initial approaches relied on standard outcome-based RL, recent works demonstrate that scaling RL on smaller architectures, as seen in DeepScaleR (Luo et al., 2025) and Skywork OpenReasoner (He et al., 2025a), can achieve performance rivaling proprietary frontiers like OpenAI-o1. To improve algorithmic stability beyond basic group-relative updates, GSPO (Zheng et al., 2025) introduces step-level granularity for precise credit assignment, whereas GMPO (Zhao et al., 2025) adopts a group-wise minimax formulation to bolster robustness against distribution shifts. Extending this to hybrid training objectives, KDRL (Xu et al., 2025b) proposes a unified framework that synergizes knowledge distillation with RL, effectively balancing teacher supervision and self-exploration. Addressing the critical balance between exploration and exploitation, the Entropy Mechanism (Cui et al., 2025) dynamically regulates policy entropy to prevent premature convergence, while LUFFY (Yan et al., 2025) improve optimization efficiency by effectively leveraging diverse, off-policy trajectories. Furthermore, emphasizing the generation of extended reasoning chains, ProRL (Liu et al., 2025a) explicitly incentivizes prolonged thought process to expand the models’ reasoning boundaries, a capability that underpins state-of-the-art large-scale systems such as MiniMax-M1 (Chen et al., 2025a).

On-policy distillation for reasoning models. Traditional knowledge distillation typically relies on offline datasets generated by a teacher model, which creates a distribution mismatch as the student’s policy drifts from the static training data (Ho et al., 2023; Hsieh et al., 2023; Ko et al., 2025b). To bridge this gap, on-policy distillation aligns the student with the teacher’s distribution by training on trajectories sampled directly from the student’s current policy (Gu et al., 2024; Agarwal et al., 2024; Ko et al., 2024). This paradigm is particularly critical for reasoning tasks, where models must learn to recover from their own logical errors rather than merely mimicking perfect teacher paths (Lu et al., 2025; Patiño et al., 2025). Recently, there has been a growing movement to adapt on-policy distillation for reasoning tasks. Frontier models like Qwen3 (Yang et al., 2025) utilize iterative on-policy feedback to refine long-chain reasoning capabilities, while MiMo-V2-Flash (Xiao et al., 2026) demonstrates that such methods achieve superior compute-efficiency by targeting “hard” examples where the student’s confidence diverges from the teacher. Furthermore, self-distillation (Zhao et al., 2026; Shenfeld et al., 2026), where the same model serves as both teacher and student and the teacher policy is conditioned on privileged information such as the ground-truth solution, has recently gained popularity.

B. Mathematical Derivations

B.1. A Fixed-credit View

The teacher-derived token credit $R_{i,t}(\theta)$ depends on θ , since it contains the student probability in the denominator. Following the stop-gradient formulation, we treat this credit as a fixed teacher-derived signal during each update. Let $\mathcal{J}_{\text{RKL}}^{\text{sg}}$ denote the corresponding objective. Then, the expected gradient satisfies

$$\nabla_{\theta} \mathcal{J}_{\text{RKL}}(\theta) = \nabla_{\theta} \mathcal{J}_{\text{RKL}}^{\text{sg}}(\theta) = \mathbb{E} \left[\frac{1}{\sum_i |o_i|} \sum_{i,t} \rho_{i,t}(\theta) R_{i,t}(\theta) \nabla_{\theta} \log \pi_{\theta}(o_{i,t} | q, o_{i,<t}) \right]. \quad (14)$$

We derive the gradient of RKL. From Eq. 1, the RKL objective and its gradient are:

$$\mathcal{J}_{\text{RKL}}(\theta) = \mathbb{E}_{q \sim Q, o \sim \pi_{\theta_{\text{old}}}(\cdot | q)} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} \rho_t(\theta) R_t(\theta) \right], \quad (15)$$

where $\rho_t(\theta) = \frac{\pi_{\theta}(o_t | q, o_{<t})}{\pi_{\theta_{\text{old}}}(o_t | q, o_{<t})}$ and $R_t(\theta) = \log \frac{\pi_T(o_t | q, o_{<t})}{\pi_{\theta}(o_t | q, o_{<t})}$. Then following holds,

$$\begin{aligned} \nabla_{\theta} \mathcal{J}_{\text{RKL}}(\theta) &= \mathbb{E}_{q \sim Q, o \sim \pi_{\theta_{\text{old}}}(\cdot | q)} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} \rho_t(\theta) \nabla_{\theta} R_t(\theta) + R_t(\theta) \nabla_{\theta} \rho_t(\theta) \right], \\ &= \mathbb{E}_{q \sim Q, o \sim \pi_{\theta_{\text{old}}}(\cdot | q)} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} \rho_t(\theta) R_t(\theta) \nabla_{\theta} \log \pi_{\theta}(o_t | q, o_{<t}) \right] - \mathbb{E}_{q \sim Q, o \sim \pi_{\theta_{\text{old}}}(\cdot | q)} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} \frac{\nabla_{\theta} \pi_{\theta}(o_t | q, o_{<t})}{\pi_{\theta_{\text{old}}}(o_t | q, o_{<t})} \right], \\ &= \mathbb{E}_{q \sim Q, o \sim \pi_{\theta_{\text{old}}}(\cdot | q)} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} \rho_t(\theta) R_t(\theta) \nabla_{\theta} \log \pi_{\theta}(o_t | q, o_{<t}) \right] - \nabla_{\theta} \mathbb{E}_{q \sim Q, o \sim \pi_{\theta_{\text{old}}}(\cdot | q)} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} \frac{\pi_{\theta}(o_t | q, o_{<t})}{\pi_{\theta_{\text{old}}}(o_t | q, o_{<t})} \right], \\ &= \mathbb{E}_{q \sim Q, o \sim \pi_{\theta_{\text{old}}}(\cdot | q)} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} \rho_t(\theta) R_t(\theta) \nabla_{\theta} \log \pi_{\theta}(o_t | q, o_{<t}) \right] \\ &\quad - \nabla_{\theta} \mathbb{E}_{q \sim Q, o \sim \pi_{\theta}(\cdot | q)} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} \frac{\pi_{\theta}(o_t | q, o_{<t})}{\pi_{\theta_{\text{old}}}(o_t | q, o_{<t})} \cdot \frac{\pi_{\theta_{\text{old}}}(o_t | q, o_{<t})}{\pi_{\theta}(o_t | q, o_{<t})} \right], \\ &= \mathbb{E}_{q \sim Q, o \sim \pi_{\theta_{\text{old}}}(\cdot | q)} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} \rho_t(\theta) R_t(\theta) \nabla_{\theta} \log \pi_{\theta}(o_t | q, o_{<t}) \right] - \nabla_{\theta} \mathbb{E}_{q \sim Q, o \sim \pi_{\theta}(\cdot | q)} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} 1 \right], \\ &= \mathbb{E}_{q \sim Q, o \sim \pi_{\theta_{\text{old}}}(\cdot | q)} \left[\frac{1}{|o|} \sum_{t=1}^{|o|} \rho_t(\theta) R_t(\theta) \nabla_{\theta} \log \pi_{\theta}(o_t | q, o_{<t}) \right] \end{aligned}$$

This justifies treating R_t as a fixed token-level credit within each update. The fixed-credit view separates how teacher credit is computed from how the resulting sampled-token credits are allocated during optimization. Our work focuses on the latter: standard RKL allocates update budget uniformly across sampled tokens, regardless of whether their credits are informative, redundant, or extreme.

B.2. Logit-Level Update-Energy Proxy

Let $p_{\theta}(\cdot | x) = \pi_{\theta}(\cdot | q, o_{<t})$ denote the student distribution at context $x = (q, o_{<t})$, and let $y = o_t$ be the sampled token. For a sampled-token objective with scalar credit $R_{i,t}(\theta)$, the token-level likelihood-ratio update with respect to the logits z is proportional to

$$g_{i,t} = R_{i,t}(\theta) \nabla_z \log p_{\theta}(y | x).$$

For a softmax distribution, we have

$$\nabla_z \log p_\theta(y | x) = e_y - p_\theta(\cdot | x),$$

where e_y is the one-hot vector of the sampled token. Therefore,

$$\|g_{i,t}\|_2^2 = R_{i,t}(\theta)^2 \|e_y - p_\theta(\cdot | x)\|_2^2.$$

Taking expectation over $y \sim p_\theta(\cdot | x)$ gives

$$\mathbb{E}_{y \sim p_\theta} [\|e_y - p_\theta(\cdot | x)\|_2^2] = 1 - \|p_\theta(\cdot | x)\|_2^2.$$

Thus, the expected squared logit-level update induced by token credit $R_{i,t}(\theta)$ is

$$\mathbb{E}_{y \sim p_\theta} [\|g_{i,t}\|_2^2] = R_{i,t}(\theta)^2 (1 - \|p_\theta(\cdot | x)\|_2^2).$$

This motivates the update-energy proxy used in Section 2:

$$E_{i,t}(\theta) = R_{i,t}(\theta)^2 (1 - \|\pi_\theta(\cdot | q, o_{i,<t})\|_2^2).$$

B.3. Softmax Score Magnitude

We prove Remark 3.1. Let $p = \text{softmax}(z)$ over vocabulary \mathcal{V} , and denote by e_y the one-hot vector for token y . For each coordinate $v \in \mathcal{V}$,

$$\frac{\partial}{\partial z_v} \log p_y = \mathbb{I}[v = y] - p_v,$$

and therefore $\nabla_z \log p_y = e_y - p$. Taking expectation over $y \sim p$ gives

$$\begin{aligned} \mathbb{E}_{y \sim p} [\|\nabla_z \log p_y\|_2^2] &= \sum_{y \in \mathcal{V}} p_y \|e_y - p\|_2^2 \\ &= \sum_{y \in \mathcal{V}} p_y (1 - 2p_y + \|p\|_2^2) \\ &= 1 - \|p\|_2^2. \end{aligned}$$

Finally, since $-\log u \geq 1 - u$ for $u \in (0, 1]$, we have $-p_v \log p_v \geq p_v(1 - p_v)$ for every v with $p_v > 0$. Summing over v and using continuity at $p_v = 0$ yields

$$H(p) = - \sum_{v \in \mathcal{V}} p_v \log p_v \geq \sum_{v \in \mathcal{V}} p_v(1 - p_v) = 1 - \|p\|_2^2.$$

B.4. Derivation of Bounded Threshold

Here, we derive the relationship between the standard log-likelihood ratio and the convex mixture ratio used to motivate our clipping threshold. Since the logarithm is a concave function, for $\forall \lambda \in [0, 1]$, Jensen's inequality implies:

$$(1 - \lambda) \cdot \log \pi_T(o_t | q, o_{<t}) + \lambda \cdot \log \pi_\theta(o_t | q, o_{<t}) \leq \log [(1 - \lambda) \cdot \pi_T(o_t | q, o_{<t}) + \lambda \cdot \pi_\theta(o_t | q, o_{<t})]. \quad (16)$$

To isolate the log-likelihood ratio $R_{i,t}(\theta) = \log \frac{\pi_T(o_t | q, o_{<t})}{\pi_\theta(o_t | q, o_{<t})}$, we subtract $\log \pi_\theta(o_t | q, o_{<t})$ from both sides and divide by $(1 - \lambda)$:

$$R_{i,t}(\theta) = \log \frac{\pi_T(o_t | q, o_{<t})}{\pi_\theta(o_t | q, o_{<t})} \leq \frac{1}{1 - \lambda} \log \frac{(1 - \lambda)\pi_T(o_t | q, o_{<t}) + \lambda\pi_\theta(o_t | q, o_{<t})}{\pi_\theta(o_t | q, o_{<t})}. \quad (17)$$

This inequality upper-bounds the log-ratio between the teacher and student policies by the log-ratio induced by a convex mixture of the two.

Comparison to Skew RKL (Ko et al., 2024). Although the RHS of Eq. (16) is identical to skew RKL (Ko et al., 2024), our approach differs in application by using the bound $\frac{\log \lambda}{1-\lambda}$ strictly as a clipping threshold rather than modifying the global objective. Figure 5 demonstrates that while Skew RKL is highly sensitive to λ (e.g., dropping significantly at $\lambda = 0.7$), our method remains robust. Remarkably, our lowest accuracy surpasses even the peak performance of Skew RKL. This confirms that selectively clipping heavy-tailed outliers stabilizes training more effectively than altering the global divergence.

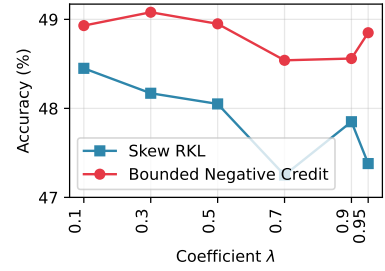


Figure 5. Sensitivity analysis of λ .

C. Full Algorithm

Here, we provide the algorithm for REOPOLD in Algorithm 1.

Algorithm 1 REOPOLD

- 1: **Input:** student policy π_θ , teacher policy π_T , query set \mathcal{Q}
 - 2: **Hyperparameters:** total steps K , switch step T_{switch} , bounding coefficient λ , top-entropy fraction β , learning rate η
 - 3: **Output:** trained student model π_θ
 - 4: Initialize $\theta_{\text{old}} \leftarrow \theta$
 - 5: **for** $k = 1$ to K **do**
 - 6: Sample a batch of queries $\mathcal{B} \sim \mathcal{Q}$
 - 7: Generate $\{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(\cdot | q)$ for each $q \in \mathcal{B}$
 - 8: Compute $R_{i,t}(\theta) \leftarrow \log \frac{\pi_T(o_{i,t}|q, o_{i,<t})}{\pi_\theta(o_{i,t}|q, o_{i,<t})}$
 - 9: Clip $\hat{R}_{i,t}^\lambda(\theta) \leftarrow \max\left(\text{sg}(R_{i,t}(\theta)), \frac{\log \lambda}{1-\lambda}\right)$ ▷ Eq. (10)
 - 10: **if** $k < T_{\text{switch}}$ **then**
 - 11: *Phase I: Exploration (Credit-based Filtering)*
 - 12: Set mask $M_{i,t}^{(k)} \leftarrow \mathbb{I}\left[R_{i,t}(\theta) \geq \frac{\log \lambda}{1-\lambda}\right]$ ▷ Eq. (12)
 - 13: **else**
 - 14: *Phase II: Refinement (Entropy-Guided Credit Allocation)*
 - 15: Compute entropy H_t^i for each token $o_{i,t}$ in batch
 - 16: Compute τ_β as the $(1 - \beta)$ -quantile of $\{H_t^i\}$
 - 17: Set mask $M_{i,t}^{(k)} \leftarrow \mathbb{I}\left[H_t^i \geq \tau_\beta\right]$ ▷ Eq. (13)
 - 18: **end if**
 - 19: Compute gradients $\nabla_\theta \mathcal{J}_{\text{REOPOLD}}$ using Eq. (4)
 - 20: Update parameters $\theta \leftarrow \theta + \eta \nabla_\theta \mathcal{J}_{\text{REOPOLD}}$
 - 21: Update old policy parameters $\theta_{\text{old}} \leftarrow \theta$
 - 22: **end for**
-

D. Detailed Experimental Setup

We list the hyperparameter values for each setting in Table 6. Sensitivity analyses for the additive hyperparameters λ and β are presented in Figure 5 and Figure 4, respectively. Regarding the switch step T_{switch} introduced in Section 3.3, we did not perform specific hyperparameter tuning; instead, it was set to approximately 1/3 of the total training steps. For K and η , we adopted the same values as our baselines, where η was determined via hyperparameter tuning based on vanilla RKL results.

Table 6. Hyperparameter values used in REOPOLD experiments in Section 4, Appendix F, and Appendix G.

Model Size	Math (Section F)		Visual (Section 4)		Agentic (Section G)
	1.5B	7B	3B	7B	3B
Total step K	300	200	60	60	40
Learning rate η	1×10^{-5}	3×10^{-6}	5×10^{-6}	5×10^{-6}	1×10^{-6}
Clipping coefficient λ	0.3	0.3	0.3	0.3	0.3
Top-entropy fraction β	0.2	0.2	0.2	0.2	0.4
Switch step T_{switch}	100	50	20	20	10

Math reasoning. We employ Verl (Sheng et al., 2025) for OPD. During rollout, we sample $n = 8$ responses per prompt with a maximum response length of 8192 and a sampling temperature of 1.0. The global batch size is set to 128 with a mini-batch size of 32, resulting in 4 gradient updates per rollout step. The student policy is trained for 300 iterations. We utilize the AdamW optimizer (Loshchilov & Hutter, 2019) with a constant learning rate of 1×10^{-5} . All training runs are conducted on a single node equipped with $8 \times$ NVIDIA H100 80GB GPUs, requiring approximately 200 and 312 GPU hours for the 1.5B and 7B models, respectively. For evaluation, we use maximum response length of 32768 and a sampling temperature of 0.6 and top-p of 0.95. Our evaluation protocol follows the setup established by Qwen2.5-Math (Yang et al., 2024) by using Verl (Sheng et al., 2025) for implementation. We employ training¹ and evaluation² data from LUFFY (Yan et al., 2025).

Visual reasoning. For visual tasks, we conduct OPD using Verl (Sheng et al., 2025). We generate $n = 12$ responses for each prompt, enforcing a maximum response length of 2048 and a sampling temperature of 1.0. Following the protocol in Liu et al. (2025b), the models are trained for 60 iterations with a batch size of 128 and a mini-batch size of 64 (equating to 2 gradient updates per step). Optimization is performed via AdamW (Loshchilov & Hutter, 2019) with a learning rate of 5×10^{-6} . Using the same hardware setup (a single $8 \times$ NVIDIA H100 node), the training takes roughly 20 and 24 GPU hours for the 3B and 7B models. For evaluation, we use maximum response length of 8192 and a sampling temperature of 0.6 and top-p of 0.95 for nucleus sampling. We utilize the training³ and evaluation⁴ data from NoisyRollout (Liu et al., 2025b).

Agentic reasoning with visual tool-use. We implement OPD and RL based on the VerlTool framework (Jiang et al., 2025). For rollout, the policy samples $n = 8$ trajectories per prompt with a maximum response length of 8192 and a temperature of 1.0. We set the maximum round for 2. We maintain a batch size of 128 and a mini-batch size of 64, corresponding to 2 gradient updates per rollout step. The student policy undergoes training for 40 iterations using the AdamW optimizer (Loshchilov & Hutter, 2019) with a constant learning rate of 1×10^{-6} . The entire process consumes approximately 120 GPU hours on a single node with $8 \times$ NVIDIA H100 80GB GPUs. Following Su et al. (2025), we set the maximum round for 5, maximum response length of 8192, sampling temperature of 1.0, and top-p of 1.0. For training, we utilize the PixelReasoner-RL dataset⁵. For evaluation, we employ the InfoVQA⁶, TallyQA⁷, and VStar⁸ datasets provided by the PixelReasoner (Su et al., 2025).

¹<https://huggingface.co/datasets/Elliott/Openr1-Math-46k-8192>

²<https://github.com/ElliottYan/LUFFY/blob/main/data/valid.parquet>

³<https://huggingface.co/datasets/xyliu6/geometry3k>

⁴https://huggingface.co/datasets/xyliu6/noisyrollout_evaluation_data

⁵<https://huggingface.co/datasets/TIGER-Lab/PixelReasoner-RL-Data>

⁶<https://huggingface.co/datasets/JasperHaozhe/InfoVQA-EvalData-PixelReasoner>

⁷<https://huggingface.co/datasets/JasperHaozhe/TallyQA-EvalData-PixelReasoner>

⁸<https://huggingface.co/datasets/JasperHaozhe/VStar-EvalData-PixelReasoner>

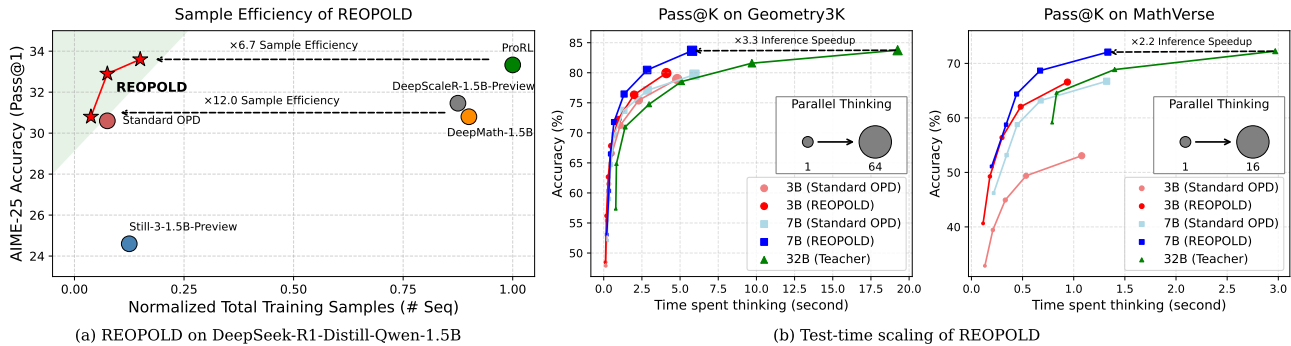


Figure 6. Performance of REOPOLD. (a) **Sample Efficiency:** REOPOLD achieves a state-of-the-art trade-off between accuracy and sample efficiency on the AIME-25 benchmark. Detailed explanation can be found in Section F. (b) **Test-time Scaling:** On visual reasoning tasks, REOPOLD demonstrates superior test-time scaling capabilities compared to the standard OPD baseline. Notably, it allows smaller models to approach the performance of the 32B teacher. Detailed explanation can be found in Section 4.

Table 7. Performance comparison of Qwen2.5-VL-3B-Instruct trained with various teacher models on a visual reasoning and perception benchmarks. Accuracy (%) is reported for all benchmarks.

	RKL	REOPOLD	$\Delta\%$
Qwen2.5-VL-7B-Instruct + NoisyRollout			
Geo3K	49.75	51.41	+1.66
MathVerse	41.66	44.27	+2.61
MathVision	23.33	24.01	+0.68
MathVista	62.20	63.10	+0.90
WeMath	64.83	65.66	+0.83
Hallusion	61.30	62.67	+1.37
AVG.	50.51	51.85	+1.34
Qwen2.5-VL-32B-Instruct + NoisyRollout[†]			
Geo3K	43.93	45.76	+1.83
MathVerse	41.99	43.22	+1.23
MathVision	25.13	25.79	+0.66
MathVista	63.60	64.20	+0.60
WeMath	64.02	64.66	+0.64
Hallusion	63.72	64.35	+0.63
AVG.	50.40	51.33	+0.93

Table 8. Performance comparison of Qwen2.5-VL-3B-Instruct trained for 60 and 300 training steps on a visual reasoning and perception benchmarks. Accuracy (%) is reported for all benchmarks.

	RKL	REOPOLD	$\Delta\%$
60 Training Steps (3B)			
Geo3K	48.09	50.58	+2.06
MathVerse	33.25	46.40	+26.4
MathVision	24.78	26.39	+3.39
MathVista	62.50	61.50	-1.60
WeMath	64.48	64.60	+0.19
Hallusion	60.83	63.62	+4.59
AVG.	48.99	51.08	+4.27
300 Training Steps (3B)			
Geo3K	49.08	51.08	+2.00
MathVerse	46.60	47.79	+1.19
MathVision	26.16	27.44	+1.28
MathVista	67.20	66.30	-0.90
WeMath	66.43	67.18	+0.75
Hallusion	65.19	66.35	+1.16
AVG.	53.44	54.36	+0.92

Table 9. Performance comparison of Qwen2.5-VL-7B-Instruct trained for 60 and 300 training steps on a visual reasoning and perception benchmarks. Accuracy (%) is reported for all benchmarks.

	RKL	REOPOLD	$\Delta\%$
60 Training Steps (7B)			
Geo3K	51.75	53.58	+3.54
MathVerse	47.71	51.43	+7.80
MathVision	28.79	29.21	+1.46
MathVista	71.27	72.40	+1.59
WeMath	70.06	69.77	-0.41
Hallusion	69.51	70.14	+0.91
AVG.	56.51	57.76	+2.21
300 Training Steps (7B)			
Geo3K	49.42	53.58	+4.16
MathVerse	50.72	51.97	+1.25
MathVision	29.74	31.12	+1.38
MathVista	71.20	73.60	+2.40
WeMath	69.43	71.84	+2.41
Hallusion	69.72	70.87	+1.15
AVG.	56.71	58.83	+2.12

E. Additional Analyses and Discussions

In this section, we provide comprehensive analyses to offer a deeper understanding of the inner workings and robustness of REOPOLD. Unless otherwise specified, all experiments follow the visual reasoning evaluation protocols introduced in Section 4.

Superior test-time scaling. We evaluate test-time scaling by measuring $Pass@K$ as a function of parallel thinking latency which is the average time to generate K responses per question. As shown in Figure 6(b), REOPOLD achieves stronger scaling curves than standard OPD and the Qwen2.5-VL-32B teacher, reaching up to $3.32\times$ higher inference efficiency in terms of $Pass@K$. This gain comes from combining higher generation quality than standard OPD with the lower latency of a compact student model. Extended $Maj@K$ results in Figure 7 show the same trend across test-time scaling metrics.

Training on different teacher. We further evaluate our approach by distilling from different teacher models to assess generalization. As detailed in Table 7, we utilize Qwen2.5-VL-7B-Instruct and 32B-Instruct models—fine-tuned on Geometry3K and MMK12 respectively via NoisyRollout (Liu et al., 2025b)—as teachers. The results demonstrate the consistent effectiveness of REOPOLD. Compared to the RKL baseline, our method yields uniform improvements across all six benchmarks for both teacher settings. Specifically, we observe an average accuracy gain of 1.34% with the 7B teacher and 0.93% with the 32B teacher. This confirms that REOPOLD is robust to variations in teacher architecture and

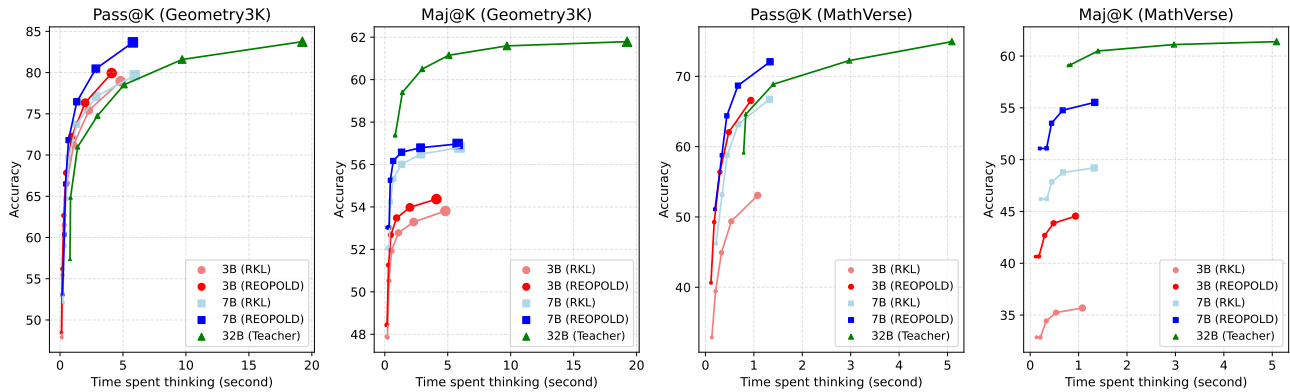


Figure 7. Extended results of Figure 6(b). We visualize the accuracy (Pass@ K and Maj@ K) against inference latency as the sample budget K increases (up to 64 for Geometry3K; 16 for MathVerse). REOPOLD (solid lines) consistently yields a better trade-off than the RKL baseline (faded lines). Notably, the 7B student matches or beats the 32B teacher’s accuracy with significantly lower latency, confirming the efficiency of our distillation.

domain-specific expertise, reliably enhancing the student’s visual reasoning and perception capabilities.

Scalability with longer training. We investigate the scalability of our approach by extending the training duration to 300 steps and integrating both Geometry3K and MMK12 datasets for both Qwen2.5-VL-3B-Instruct (Table 8) and Qwen2.5-VL-7B-Instruct (Table 9). As shown in Table 8 and Table 9, extending the training horizon yields performance gains for both the baseline and our method, confirming the benefit of larger-scale training. While longer training generally improves performance across the board, REOPOLD consistently demonstrates better scalability. It outperforms the RKL baseline in both model sizes, achieving the highest average accuracy of 54.36% with the 3B model and 58.83% with the 7B model. These results indicate that our method is capable of continuously refining its policy given more compute and data, leading to robust improvements in both visual reasoning and perception tasks.

Extended test-time scaling results. We provide a comprehensive evaluation of the test-time scaling capabilities of REOPOLD on Geometry3K (Lu et al., 2021) and MathVerse (Zhang et al., 2024a) benchmarks. Figure 7 demonstrates that the superior scaling of REOPOLD is not limited to coverage metrics (Pass@ K). Both the 3B and 7B models maintain a consistent lead in Maj@ K , a metric that measures consensus robustness. This confirms that our method fundamentally increases the probability of correct reasoning chains, rather than merely generating over diverse “lucky guess” to boost Pass@ K .

Training time analysis. We analyze training wall-clock time to quantify computational overhead as shown in Figure 8. Contrary to concerns about the teacher’s cost in on-policy distillation, our breakdown reveals it accounts for only a moderate fraction (8%–22%) of the total step. Crucially, this relative cost is inversely proportional to generation length. In long-context math tasks (8192 tokens), student generation dominates the runtime (77.3%), rendering the teacher’s impact marginal (8.2%). Even in shorter visual reasoning tasks (2048 tokens), where the teacher’s share rises to 21.8%, the primary bottleneck remains the student’s generation process rather than teacher supervision.

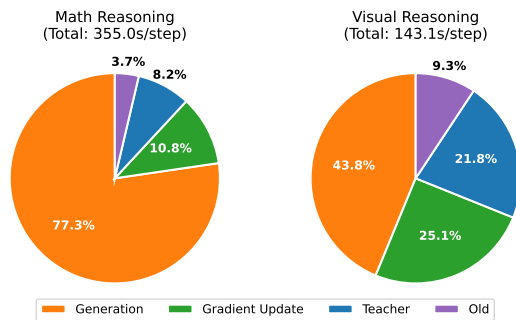


Figure 8. Breakdown of training wall-clock time per step.

Extension to Top-K OPD. We further compare REOPOLD with GKD (Agarwal et al., 2024) and DistiLLM (Ko et al., 2024) which are full-vocabulary OPD. As shown in Table 10, both GKD and DistiLLM with full vocabulary incur out-of-memory (OOM) issue since they need to store the value for 150K vocab size for both student and the teacher models. To alleviate this cost, we apply the commonly used Top-K ($K = 5$) approximation (Li et al., 2026); however, this approximation is less effective and yields lower efficiency than sampled-token approaches. Nevertheless, under the same Top-5 setting, REOPOLD still outperforms Top-5 approximated GKD and DistiLLM, showing that its gains are not limited to sampled-token training. Importantly, applying either full-vocabulary scoring or the Top-5 approximation to REOPOLD does not bring meaningful improvement over sampled-token REOPOLD, suggesting that the proposed credit allocation strategy is already sufficient without costly vocabulary-level approximation.

Table 10. Comparison with full-vocabulary OPD and Top-K ($K = 5$) approximations. Full-vocabulary GKD and DistiLLM (DLM) runs incur OOM under our visual reasoning setup; therefore, we additionally report Top-5 approximations.

	GKD/DLM	GKD [†] (Agarwal et al., 2024)	DLM [†] (Ko et al., 2024)	REOPOLD	REOPOLD [†]
Geo3K	OOM	51.85	52.68	53.58	53.41
MathVerse	OOM	48.14	48.65	51.97	50.96
MathVision	OOM	29.90	30.12	31.12	30.94
MathVista	OOM	71.25	72.12	73.60	73.64
WeMath	OOM	70.87	70.92	71.84	71.80
Hallusion	OOM	70.21	70.37	70.87	71.52
AVG.	OOM	57.04	57.48	58.83	58.71

Table 11. Performance comparison of REOPOLD and distillation baselines (SFT and standard OPD) on mathematical reasoning benchmarks across different teacher models. Accuracy (%) is reported for all benchmarks. The best result in each column is shown in **bold**, and the second-best is underlined. The results are reported as mean \pm standard deviation over 5 runs.

Model	AIME-24	AIME-25	AMC-23	MATH-500	Minerva Math	Olympiad Bench	AVG.
SkyWork-OR1-Math-7B (He et al., 2025a) \rightarrow DeepSeek-R1-Distill-Qwen-1.5B (Guo et al., 2025)							
SkyWork-OR1-Math-7B	69.8	52.3	94.1	95.8	49.3	73.5	72.5
R1-Distill-Qwen-1.5B	28.6	22.7	62.6	82.9	26.4	43.6	44.4
+ GRPO	31.8 \pm 1.0	23.7 \pm 0.9	62.0 \pm 1.1	85.4 \pm 0.6	33.8 \pm 0.3	49.8 \pm 0.6	47.8 \pm 0.3
+ SFT	33.5 \pm 1.5	24.6 \pm 1.9	76.1 \pm 1.2	86.6 \pm 0.8	<u>36.4</u> \pm 0.8	55.6 \pm 0.8	51.5 \pm 0.9
+ Standard OPD	<u>37.1</u> \pm 1.4	<u>30.6</u> \pm 0.6	<u>80.2</u> \pm 0.9	<u>88.0</u> \pm 0.5	34.6 \pm 0.4	<u>56.0</u> \pm 0.8	<u>54.4</u> \pm 0.2
+ REOPOLD	41.6 \pm 1.1	32.6 \pm 0.4	83.0 \pm 1.2	89.2 \pm 0.4	38.6 \pm 0.7	57.3 \pm 0.5	57.1 \pm 0.3
Phi-4-Mini-Reasoning (Xu et al., 2025a) \rightarrow DeepSeek-R1-Distill-Qwen-1.5B (Guo et al., 2025)							
Phi4-Mini-Reasoning	55.6	41.4	91.8	94.6	39.1	64.5	64.5
+ Standard OPD	13.3 \pm 0.5	21.0 \pm 1.1	45.5 \pm 2.1	78.6 \pm 2.7	19.4 \pm 1.4	32.8 \pm 1.0	35.1 \pm 1.1
+ REOPOLD	32.1 \pm 1.7	24.6 \pm 0.7	65.2 \pm 1.2	83.7 \pm 1.3	30.2 \pm 0.5	45.9 \pm 0.6	47.0 \pm 0.4

F. Extension: Math Reasoning

Setup. We conduct on-policy distillation on DeepSeek-R1-Distill-Qwen-1.5B and 7B (Guo et al., 2025), employing SkyWork-OR1-7B (or Phi-4-mini-reasoning (Xu et al., 2025a)) and SkyWork-OR1-32B-Preview (He et al., 2025a) as teachers, respectively. For training, we utilize the dataset proposed by Yan et al. (2025), which contains 45k prompts. While all 1.5B models in Table 11 are trained for 300 steps for fair comparison, we extend the training of REOPOLD to 600 steps for the sample efficiency analysis in Figure 6(a). Detailed training setup is described in Appendix D.

Evaluation. We evaluate all models on six competition-level mathematical reasoning benchmarks: AIME-24, AIME-25, AMC-23, MATH-500 (Hendrycks et al., 2020), Minerva Math (Lewkowycz et al., 2022), and Olympiad Bench (He et al., 2024). For AIME-24, AIME-25, and AMC-23, we report Avg@32 to ensure robust evaluation considering the relatively small test sets. For the remaining three benchmarks, we report Pass@1. In all evaluations, we use a temperature of 0.6 and a top-p value of 0.95.

Result 1: Better training sample-efficiency. As shown in Figure 6(a), REOPOLD matches ProRL (Liu et al., 2025a) in 600 steps (vs. 2000); when normalized by the total number of generated sequences, computed as steps \times batch size \times rollouts, this yields a $> 6.7\times$ efficiency gain. It surpasses DeepScaleR-1.5B-Preview (Luo et al., 2025) and DeepMath-1.5B (He et al., 2025b) even earlier at 300 steps ($> 12\times$ efficiency), and notably outperforms standard OPD (300 steps) in just 150 steps. Finally, superior performance over GRPO in our re-implementation under identical conditions (Table 11) confirms that our gains stem from algorithmic efficacy rather than experimental setup.

Result 2: Robustness to different model family. Using the sequence alignment (Patiño et al., 2025; Boizard et al., 2025), we distill Phi-4-Mini-Reasoning into DeepSeek-R1-Distill-Qwen-1.5B, where the teacher and student come from different model families and use different vocabularies. This setting induces a large teacher-student distribution mismatch, under which standard OPD suffers severe negative transfer, reducing average performance from 44.4 to 35.1. In contrast, REOPOLD improves the student to 47.0, showing that selective credit allocation can stabilize OPD even when teacher feedback is transferred across mismatched vocabularies and model families.

Result 3: Scaling to large policy models. We further evaluate REOPOLD on a stronger student, DeepSeek-R1-Distill-Qwen-7B. As shown in Figure 9, standard OPD is unstable: performance drops sharply early in training and fails to recover beyond the base model on AIME-24. In contrast, REOPOLD prevents early degradation and steadily improves across benchmarks, demonstrating that selective credit allocation remains effective when scaling OPD to stronger policy models.

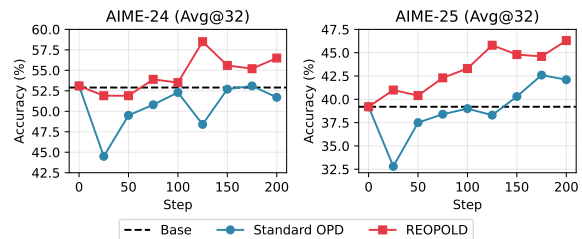


Figure 9. Scaling to a 7B student exposes instability in standard OPD, while REOPOLD maintains stable and consistent improvement.

Table 12. Performance comparison of vision-language models on agentic visual tool-use tasks. Accuracy (%) is reported for all benchmarks. The best and the second-best results in each column are shown in **bold** and underlined, respectively.

Model	Pixel	V-Star	InfoVQA	TallyQA	AVG.
Pixel-Reasoner-7B	64.00	84.29	74.37	73.69	74.05
Pixel-Reasoner-7B (Su et al., 2025) → Qwen2.5-VL-3B-Instruct + SFT (Jiang et al., 2025)					
Qwen2.5-VL-3B-Instruct + SFT	46.00	71.20	34.88	56.91	45.83
+ GRPO (Jiang et al., 2025)	60.00	<u>76.96</u>	59.47	60.56	<u>64.25</u>
+ standard OPD	52.00	76.55	<u>61.09</u>	<u>64.34</u>	63.27
+ REOPOLD	<u>57.00</u>	77.43	63.12	65.43	65.75

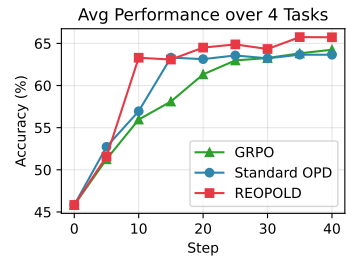


Figure 10. Average score by training step. REOPOLD outperforms all baselines at 50% training.

G. Extension: Agentic Reasoning with Visual Tool-Use

Traditional visual reasoning approaches typically treat images as static inputs, limiting the model’s ability to actively explore visual information. To address this limitation, we implement image operation tools that enable agents to zoom into specific regions, select key frames, and perform other visual manipulations. This approach, following Pixel-Reasoner (Su et al., 2025), enhances reasoning capabilities over dense visual data.

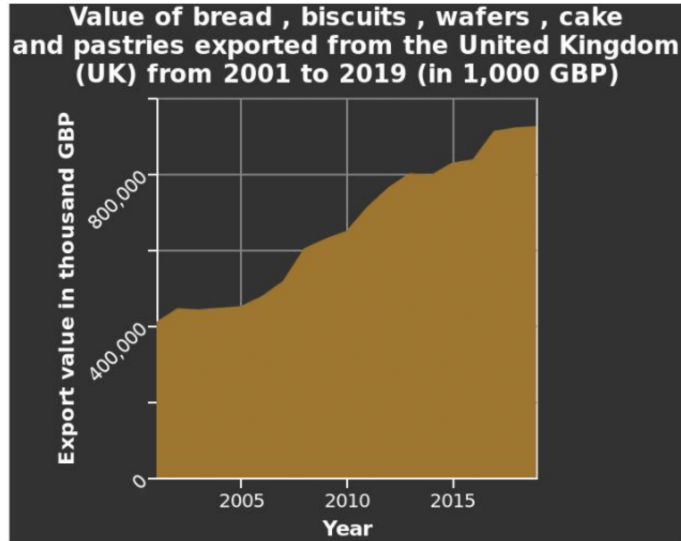
Setup. We implement our proposed method based on the VerlTool (Jiang et al., 2025) framework. We adopt the SFTed Qwen2.5-VL-3B-Instruct (Jiang et al., 2025) as the student policy and Pixel-Reasoner-7B (Su et al., 2025) as the teacher. We use the official training dataset from Pixel-Reasoner, comprising 15K queries from InfographicVQA, supplemented by additional public datasets. Detailed training setup is described in Appendix D.

Evaluation. Following Su et al. (2025), we evaluate our model and baselines on four representative multi-modal benchmarks using nucleus sampling with temperature of 1.0 and a top-p of 1.0: test-split of Pixel-Reasoner, V-Star (Wu & Xie, 2024), InfographicVQA (Mathew et al., 2022), and TallyQA (Acharya et al., 2019). This selection offers a wide spectrum of visual understanding tasks, ranging from fine-grained object recognition to high-level reasoning in both static and dynamic scenarios.

Results. As reported in Table 12, REOPOLD outperforms both standard OPD and GRPO, notably surpassing the latter even when it utilizes the complex reward designs proposed by Su et al. (2025). Although GRPO achieves slightly higher accuracy on the Pixel test split, REOPOLD demonstrates superior performance across other benchmarks, indicating stronger generalization capabilities. Furthermore, we show in Figure 10 that REOPOLD exhibits better sample efficiency compared to standard OPD and GRPO. Unlike traditional RL approaches that necessitate intricate reward engineering for sophisticated agentic tasks, REOPOLD can be applied directly.

H. Qualitative Evaluation

Question: From 2001 to 2010, Is 2010 exported the most value of bread, biscuits, wafers, cakes and pastries from the UK? Your final answer can only be *yes* or *no*.



[Qwen2.5-VL-3B-Instruct (REOPOLD)]

<think>

To determine if 2010 exported the most value... we need to analyze the graph step by step.

1. **Identify the Time Period:** The graph shows data from 2001 to 2019 ...
2. **Analyze the Graph:** The graph is a line graph ...
6. **Conclusion:** Based on the graph, **the export value in 2010 is the highest** ...

</think>

Final Answer: *yes*

[Qwen2.5-VL-3B-Instruct (Standard OPD)]

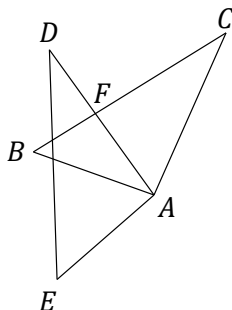
Step-by-Step Reasoning:

1. **Identify the Time Period:** The question asks about ...
4. **Determine the Maximum Value:** ... **values are relatively close to 400,000 GBP for each year from 2001 to 2010. There is no clear indication that 2010 had a significantly higher** ...
5. **Conclusion:** Therefore, it is not possible to definitively say ...

Final Answer: *No*

Figure 11. **Qualitative comparison on Hallusion Bench.** While the baseline trained with Standard OPD suffers from visual perception degradation (hallucinating a flat trend despite the visual evidence), REOPOLD maintains robust visual grounding, accurately identifying the peak in the chart.

Question: As shown in the figure, angle C = 36.0, angle BAE = 60.0, then the degree of angle AFC is ()
 Choices: A:84° B:80° C:60° D:90°



[Qwen2.5-VL-7B-Instruct (REOPOLD)]

<think>

To determine the measure of angle AFC, let's analyze the given information...

4. **Analyze the angle AFC:**

... Substituting the given values:

$$\text{Angle AFC} = 60^\circ + 36^\circ = 96^\circ$$

However, we need to re-evaluate the problem because the angle AFC is actually the angle formed by the intersection...

5. **Re-evaluate using the properties of the angles:**

... The correct measure of angle AFC is:

$$\text{Angle AFC} = 180^\circ - 60^\circ - 36^\circ = 84^\circ$$

</think>

A

[Qwen2.5-VL-7B-Instruct (Standard OPD)]

<think>

To determine the measure of angle AFC...

5. **Consider the angle BAE:**

... Therefore:

$$60^\circ = x + (144^\circ - x)$$

This equation is not directly useful for finding x , but it confirms the relationship...

...

8. **Final calculation:**

The correct measure of angle AFC is:

$$\text{Angle AFC} = 156^\circ - 60^\circ$$

$$\text{Angle AFC} = 90^\circ$$

</think>

90

Figure 12. **Qualitative comparison on MathVerse.** The baseline (Standard OPD) gets trapped in circular logic (Step 5) and hallucinates the final calculation (Step 8). Our model initially derives an incorrect value (96°) but explicitly triggers a self-correction process ("However, we need to re-evaluate...") to reach the correct solution (84°).

I. Limitations & Impact Statement

Limitations. A limitation of REOPOLD is that its gains may be smaller when the teacher and student are already closely matched, since most token-level credits become near zero and standard OPD is less unstable in this regime. In addition, REOPOLD improves the allocation and stability of teacher feedback but does not directly reduce the cost of teacher inference; when teacher evaluation dominates wall-clock training time, the practical speedup may be limited. Future work could adapt the allocation rule to the teacher-student gap and combine selective credit allocation with cheaper teacher scoring or caching strategies.

Broader Impact. This work improves the efficiency and stability of reasoning distillation for compact language models. By making sampled-token OPD more effective across mathematical, visual, and agentic tool-use tasks, REOPOLD may reduce the data and optimization cost required to train capable smaller models, lowering deployment cost and latency in resource-constrained settings. However, stronger compact models may also lower the barrier to misuse, particularly in automated or tool-use applications. These risks are not unique to REOPOLD and should be mitigated through safety evaluation, access control, and deployment monitoring.

J. Reproducibility Statement

For our experiments, we have clearly described the datasets used, the models fine-tuned, and the hyperparameters used (Section F, Section 4, Section G, and Appendix D). While all datasets and base models are publicly available, we provide a summary in Appendix D, along with relevant links. To further aid reproducibility, we provide a standalone implementation of our core algorithm in the supplementary material. We will provide full source code upon acceptance.