Anti-Self-Distillation for Reasoning RL via Pointwise Mutual Information

Guobin Shen[1] **Xiang Cheng**[1] **Chenxiao Zhao**[1] **Lei Huang**[1]
Jindong Li[2] **Dongcheng Zhao**[2] **Xing Yu**[1*]

[1]Xiaohongshu Inc. [2]Institute of Automation, Chinese Academy of Sciences

Abstract

On-policy self-distillation, where a student is pulled toward a copy of itself conditioned on privileged context (e.g., a verified solution or feedback), offers a promising direction for advancing reasoning capability without a stronger external teacher. Yet in math reasoning the gains are inconsistent, even when the same approach succeeds elsewhere. A pointwise mutual information analysis traces the failure to the privileged context itself: it inflates the teacher's confidence on tokens already implied by the solution (structural connectives, verifiable claims) and deflates it on deliberation tokens (*Wait*, *Let*, *Maybe*) that drive multi-step search. We propose Anti-Self-Distillation (AntiSD), which ascends a divergence between student and teacher rather than descending it: this reverses the per-token sign and yields a naturally upper-bounded advantage in one step. An entropy-triggered gate disables the term once the teacher entropy collapses, completing a drop-in replacement for default self-distillation. Across five models from 4B to 30B parameters on math reasoning benchmarks, AntiSD reaches the GRPO baseline's accuracy in 2 to $10\times$ fewer training steps and improves final accuracy by up to 11.5 points.

`github.com/FloyedShen/AntiSD` `wandb.ai/brain-cog/AntiSD`

1 Introduction

Reinforcement learning has become a primary axis of post-training progress for reasoning tasks, with reinforcement learning from verifiable rewards (RLVR; 24; 34; 5; 10) emerging as the dominant paradigm. The reward signal in RLVR, however, is typically a sparse, trajectory-level scalar: a single bit per rollout that does not indicate which intermediate step was responsible, leaving credit assignment to individual reasoning steps as an open problem. To address this, two main directions have emerged: training a separate process reward model (PRM) to score intermediate steps [15; 27; 19], or applying on-policy distillation (OPD) to provide a token-level imitation signal from a stronger teacher [1; 4; 18]. Both, however, depend on an external model. Can the model itself supply this credit?

On-policy self-distillation answers this in the affirmative. It specializes OPD by taking the teacher to be the student itself, conditioned on privileged context: typically a verified solution and any feedback from the environment. The token-level signal is then produced by the model's own forward pass under richer conditioning, requiring neither an external teacher nor a separate reward model. A series of recent works [38; 7; 33; 22] has developed this idea along several axes, connecting back to the older framework of learning under privileged information [26; 17].

On math reasoning, however, the picture is more mixed. Diagnostic studies report that on-policy self-distillation can improve instruction-following, scientific QA, and tool-use tasks [7], while delivering only modest or inconsistent gains on more challenging mathematical problems [9]. We observe the same pattern across model families ranging from 4B to 30B parameters: on math reasoning

*Correspondence to: `yuanshan2@xiaohongshu.com`, `floyed_shen@outlook.com`.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

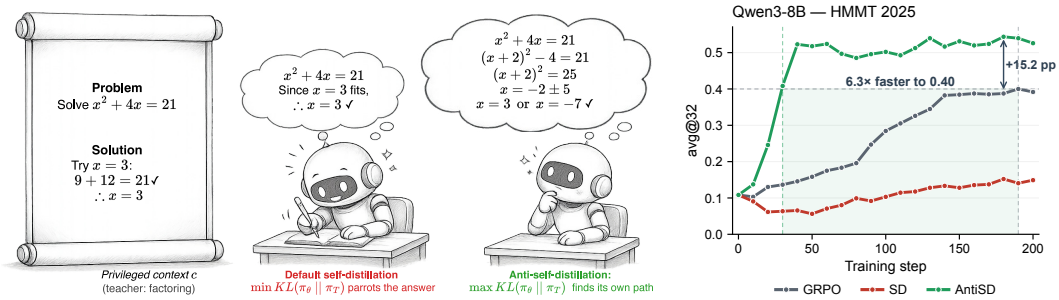


Figure 1: **(a)** An oracle-conditioned teacher biases the student toward a single root; reversing the signal preserves deliberation and recovers the rest. **(b)** Qwen3-8B on HMMT 2025 (avg@32). Dashed lines: first steps reaching 0.40 (a benchmark-specific $6.3\times$ speedup, distinct from the five-benchmark $5.0\times$ of Table 1). At the Table 1 best-Avg checkpoints AntiSD leads GRPO by $+15.2$ pp (54.4 vs 39.2); default self-distillation underperforms GRPO.

benchmarks such as AIME 2024 and 2025, default self-distillation typically fails to outperform a strong GRPO baseline (Figure 1 (b) shows one representative case; full sweep in Section 4.1).

To understand the cause, we inspect the per-token signal that default self-distillation produces (Figure 2). The pattern points to the privileged context itself: conditioning the teacher on a verified solution effectively turns it into an oracle, leaving it confident on tokens that follow once the answer is known, such as structural connectives and verifiable-claim words, and unsure on deliberation tokens like *Wait*, *Let*, and *Maybe* that the student emits when re-examining alternatives. Standard self-distillation pulls the student toward this oracle teacher, reinforcing tokens that track the known solution and weakening tokens that drive deliberation, as shown in Figure 1 (a).

This motivates a simple fix: invert the gradient direction. We propose *Anti-Self-Distillation* (AntiSD), which ascends a divergence between student and teacher rather than descending it, reversing the per-token sign and yielding a naturally upper-bounded advantage in one step. An entropy-triggered gate disables the term once the teacher's per-token entropy collapses, completing a drop-in replacement for default self-distillation. Across five models from 4B to 30B parameters on math reasoning benchmarks, AntiSD reaches the GRPO baseline's accuracy in 2 to $10\times$ fewer training steps and improves final accuracy by up to 11.5 points.

Our contributions are summarized as follows:

- We expose a structural shortcut bias in standard self-distillation, where the per-token signal rewards tokens the privileged context already implies and suppresses deliberation tokens, and ground this observation in a conditional pointwise mutual information identity (Section 3.1).
- We propose *Anti-Self-Distillation* (AntiSD), which reverses the per-token signal by ascending Jensen-Shannon divergence between student and teacher; the JSD shape bounds the advantage on the deliberation side, and an entropy-triggered gate disables the term once the teacher entropy collapses. AntiSD is a drop-in replacement for default self-distillation with no additional cost.
- We evaluate on math reasoning, with an initial code probe on Qwen3-8B. Across five 4B–30B models on math, AntiSD matches the GRPO baseline in 2 to $10\times$ fewer steps and improves final accuracy by up to 11.5 points over GRPO; default self-distillation stays below GRPO on all five models, as does a matched OPSD baseline on Qwen3-8B.

2 Preliminaries

Setup. We work with an autoregressive language model π_θ that, given a problem x, samples a trajectory $y = (y_1, \ldots, y_T)$. RLVR provides a scalar verifiable reward $R(x, y)$ scoring the final answer. Following GRPO [24], we sample a group of G rollouts per prompt and use the group-normalized sequence-level advantage $A_i^{\text{seq}} := (R_i - \mu_R)/\sigma_R$ as the policy-gradient signal for the i-th rollout, where μ_R and σ_R are the within-group mean and standard deviation.

On-policy self-distillation. On-policy self-distillation augments the GRPO objective with a per-token signal derived from a self-teacher. Let c denote privileged context (in our setting, an outcome-verified peer solution when one exists, plus correctness feedback; Section 4) provided at training time but

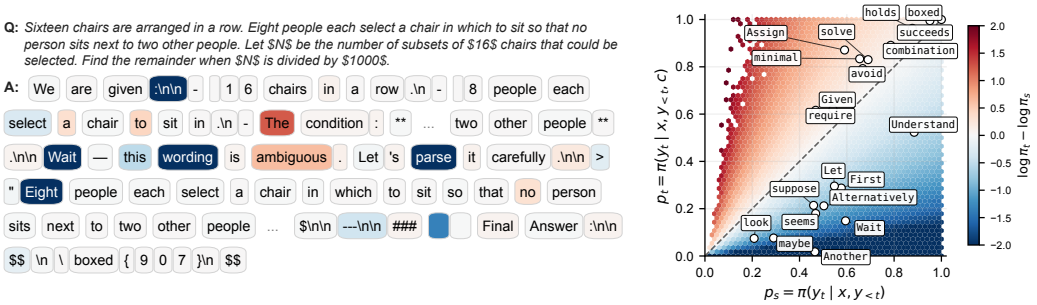


Figure 2: Per-token signal $u_t = t_t - s_t$ on Qwen3-4B-IT-2507 at AIME-25. **(a)** Single-rollout trace. **(b)** (π_S, π_T) heatmap. Blue marks deliberation tokens ($u_t \ll 0$); red marks shortcut tokens ($u_t > 0$).

not at inference. The same network π_θ plays two roles: the student $\pi_S(\cdot \mid x, y_{<t}) := \pi_\theta(\cdot \mid x, y_{<t})$ generates the rollout, while the teacher $\pi_T(\cdot \mid x, y_{<t}) := \pi_\theta(\cdot \mid x, c, y_{<t})$ scores it under richer conditioning (we suppress c from the teacher's left-hand-side conditioning as a notational shorthand, since c is fixed throughout each training step). With $\mathrm{sg}[\cdot]$ denoting stop-gradient, the standard self-distillation loss is the per-token KL,

$$\mathcal{L}_{\mathrm{SD}}(\theta) \;=\; \mathbb{E}_{x,\, y\sim\pi_S(\cdot|x)}\Big[\sum_{t=1}^{T} D_{\mathrm{KL}}\big(\pi_S(\cdot \mid x, y_{<t}) \,\big\|\, \mathrm{sg}[\pi_T(\cdot \mid x, y_{<t})]\big)\Big], \tag{1}$$

in addition to the GRPO objective. More generally, $\mathcal{L}_{\mathrm{SD}}$ is one member of a family of per-token f-divergences between student and teacher; the choice of f shapes the resulting per-token advantage and we revisit it in Section 3.2. The basic on-policy distillation formulation drops the GRPO term ($A_i^{\mathrm{seq}} \equiv 0$) and uses this per-token signal alone [1; 4; 38]; recent reasoning RL methods [7; 13; 29] instead combine it with the trajectory-level reward A_i^{seq} through various forms (additive, multiplicative, or sample-level routing). We adopt the additive form:

$$A_{i,t} \;=\; A_i^{\mathrm{seq}} \;+\; \lambda \cdot \delta_t, \tag{2}$$

where δ_t is the per-token contribution of $-\nabla_\theta \mathcal{L}_{\mathrm{SD}}$ written in policy-gradient form (closed form in Section 3.1) and $\lambda \geq 0$ is a mixing weight.

3 Anti-Self-Distillation

Section 3.1 identifies the per-token signal δ_t from Equation (2) with conditional pointwise mutual information and shows, in conjunction with Figure 2, that it carries a structural shortcut bias. Section 3.2 responds with *Anti-Self-Distillation* (AntiSD), which ascends Jensen-Shannon divergence between student and teacher under a single entropy-triggered gate.

3.1 Per-token reward as conditional PMI

We abbreviate $s_t := \log \pi_S(y_t \mid x, y_{<t})$, $t_t := \log \pi_T(y_t \mid x, y_{<t})$, and $u_t := t_t - s_t$. To get a closed form for δ_t in Equation (2), differentiate the per-token KL summand $\mathbb{E}_{v\sim\pi_S}[\log \pi_S(v) - \log \pi_T(v)]$ in Equation (1) with respect to θ. The constant-coefficient term $\mathbb{E}_v[\nabla_\theta \log \pi_S(v)]$ vanishes by the score-function identity $\sum_v \nabla_\theta \pi_S(v) = 0$, the teacher gradient is killed by the stop-gradient, and only a weighted score-function term survives (full proof in Appendix A, Lemma 1):

$$\nabla_\theta\, D_{\mathrm{KL}}\big(\pi_S(\cdot \mid x, y_{<t}) \,\big\|\, \pi_T(\cdot \mid x, y_{<t})\big) \;=\; -\,\mathbb{E}_{v\sim\pi_S(\cdot|x,y_{<t})}\Big[\, u_v \cdot \nabla_\theta \log \pi_S(v \mid x, y_{<t}) \,\Big], \tag{3}$$

with $u_v := \log \pi_T(v \mid x, y_{<t}) - \log \pi_S(v \mid x, y_{<t})$. The combined advantage in Equation (2) therefore uses $\delta_t = +u_t$. Following standard policy-gradient practice for distillation, we treat the outer rollout expectation $\mathbb{E}_{y\sim\pi_S}$ as a sample-mean estimator with stop-gradient on the trajectory distribution; the trajectory-level REINFORCE term that would otherwise arise from differentiating $\pi_S(y \mid x)$ is dropped, since trajectory-level credit assignment is handled separately by the GRPO term A_i^{seq}.

u_t as conditional PMI. Under the self-distillation setup, π_S and π_T share parameters, so u_t admits a closed-form interpretation:

$$u_t \;=\; \log \frac{\pi_\theta(y_t \mid x, c, y_{<t})}{\pi_\theta(y_t \mid x, y_{<t})} \;=\; \mathrm{PMI}(y_t\,;\, c \mid x, y_{<t}), \tag{4}$$

the conditional pointwise mutual information between the next token y_t and the privileged context c. The sign of u_t records whether c raises ($u_t > 0$) or lowers ($u_t < 0$) $\pi_\theta(y_t)$. The default per-token reward $\delta_t = +u_t$ therefore rewards tokens whose probability is raised by c and penalizes those it lowers; Figure 2 makes this concrete on real data.

We compute u_t on student rollouts from Qwen3-4B-IT-2507 at AIME-25, with c from our self-distillation pipeline (Appendix C). The teacher reward splits tokens into two informative regimes. *Shortcut tokens* ($u_t > 0$, red) – *Given*, *Assign*, *succeeds*, *holds* – are rewarded once the answer is known. *Deliberation tokens* ($u_t \ll 0$, deep blue) – *Wait*, *Let*, *Maybe*, *Alternatively* – are strongly penalized, since c has committed to a solution and the teacher down-weights tokens that re-examine alternatives. Generic tokens along the diagonal and answer-template tokens near $(\pi_S, \pi_T) \approx (1, 1)$ carry no signal. Figure 2(a) traces these regimes alternating along a single rollout, and (b) aggregates them into a (π_S, π_T) heatmap with two off-diagonal lobes of opposite u_t sign.

Default self-distillation thus rewards shortcut tokens and penalizes deliberation tokens. This is consistent with a phenomenon repeatedly observed under on-policy self-distillation – responses shorten as training proceeds [7; 9; 22] – but recasts it as a structural shortcut rather than benign compression, with the suppression concentrated on the deliberation steps that drive search rather than on redundant filler. The polarity is not specific to reverse KL: for any convex f in the family from Section 2, descent on $D_f(\pi_S \| \pi_T)$ has a per-token advantage that is non-decreasing in u_t and vanishes at $u_t = 0$ (Lemma 6), so it inherits the same shortcut/deliberation split.

Two empirical observations from this analysis will drive the method. *(O1) Wrong polarity for reasoning*: the per-token reward $\delta_t = +u_t$ has the wrong sign – rewarding shortcut tokens and penalizing the deliberation tokens that drive search. *(O2) Asymmetric distribution*: rollouts are sampled from π_S, so the $u_t < 0$ tokens ($\pi_S > \pi_T$) are over-sampled relative to teacher sampling at the same context, by exactly $\pi_S(V^-) - \pi_T(V^-) = \mathrm{TV}(\pi_S, \pi_T) \geq 0$ for $V^- = \{v : \pi_S(v) > \pi_T(v)\}$ (Remark 7), though they need not form a batch majority; that they also form the heavier lobe in Figure 2(b) is a separate empirical observation. The log-ratio $u_t = \log[\pi_T(y_t)/\pi_S(y_t)]$ can be arbitrarily negative; in Figure 2(a), individual tokens the student samples but the teacher nearly rules out reach $u_t \leq -20$ ($\pi_T/\pi_S \leq e^{-20}$).

3.2 Ascent on Jensen-Shannon divergence

AntiSD has three components. From (O1), we *reverse the gradient direction* (descent $\rightarrow$ ascent), flipping the per-token reward at the source. From (O2), we *ascend Jensen-Shannon divergence* rather than reverse KL: JSD's f-divergence-derived advantage is asymmetrically bounded (capped on the over-sampled deliberation side and linear on the under-sampled shortcut side), directly counterbalancing the empirical asymmetry. The third component, an *entropy-triggered gate*, follows from the first two: once we ascend a divergence, the policy gradient is no longer self-terminating, so we need a signal-quality criterion to disable the term once π_T's information about π_S degenerates. We make each concrete below.

JSD ascent. Writing $D_{\mathrm{JSD}}(\pi_S \| \pi_T) = \mathbb{E}_{\pi_T}[f(\pi_S/\pi_T)]$ for the corresponding f-divergence generator, the score-function trick (analogous to Equation (3)) gives

$$\nabla_\theta D_{\mathrm{JSD}}(\pi_S \| \pi_T) \;=\; \mathbb{E}_{v \sim \pi_S}\left[f'\left(\tfrac{\pi_S(v)}{\pi_T(v)}\right) \nabla_\theta \log \pi_S(v) \right]. \tag{5}$$

Substituting $\pi_S/\pi_T = e^{-u}$ identifies $f'(\pi_S/\pi_T) = -\varphi(u)$ (full simplification in Appendix A), so ascending JSD via policy gradient has per-token advantage

$$A_t^{\mathrm{AntiSD}} \;=\; -\varphi(u_t), \qquad \varphi(u) \;:=\; \tfrac{1}{2}(\mathrm{softplus}(u) - \log 2)\,. \tag{6}$$

The shape φ is the f-divergence derivative for D_{JSD}, so its monotonicity and sign-preservation follow from JSD's convexity. At small $|u|$, $\varphi'(0) = \frac{1}{2}\sigma(0) = \frac{1}{4}$ gives $-\varphi(u) \approx -\frac{1}{4}u$, which recovers ascent on reverse KL up to a positive scalar; the two choices diverge in the tails, where $\varphi(u) \geq -\frac{1}{2}\log 2$ globally (proof in Appendix A) caps the AntiSD advantage on the deliberation side at $\frac{1}{2}\log 2$. This is exactly the side that (O2) flagged as both over-sampled and heavy-tailed: the cap absorbs the $u_t \leq -20$ spikes and rebalances per-token gradient contributions against the lighter, under-sampled shortcut side, while the shortcut side keeps its linear penalty since extreme shortcut tokens are precisely the ones AntiSD should suppress proportionally. We ablate the divergence choice in Section 4.3.

Entropy-triggered gate. The JSD ascent direction is not self-terminating, so we need a criterion to disable the term once u_t stops carrying useful conditional information. The teacher's per-token entropy aggregated over the batch, $H := \mathrm{median}_{i,t}\, H[\pi_T(\cdot \mid x_i, y_{i,<t})]$, provides this signal. The log-ratio $u_t = \log(\pi_T/\pi_S)$ is well-conditioned only as long as π_T retains substantial entropy: when π_T collapses to a near-deterministic mode (low H), most tokens lie at floor probability under π_T and u_v becomes dominated by numerical floor rather than conditional information. We disable the AntiSD term when H falls below an auto-calibrated threshold τ_{down}, and re-enable it once H recovers to its pre-collapse baseline H_{warm} (a Schmitt trigger to avoid chatter):

$$g \leftarrow \begin{cases} 1 & \text{if } g = 0 \text{ and } H \geq H_{\mathrm{warm}}, \\ 0 & \text{if } g = 1 \text{ and } H < \tau_{\mathrm{down}}, \\ g & \text{otherwise,} \end{cases} \qquad \lambda = g \cdot \lambda_{\max}. \tag{7}$$

τ_{down} is auto-calibrated from W warmup steps at $\lambda = 0$ (concrete values in Section 4 Setup). Algorithm 1 (Appendix B) summarizes the resulting update.

4 Experiments

Setup. We train five language models from the Qwen3 [31] and Olmo-3 [21] families (4B–30B parameters) on DAPO-Math-17k [34] for 200 on-policy steps, comparing four conditions per model: the un-trained base, +GRPO (Equation (2) with $\lambda = 0$), +SD (default self-distillation: the sampled-token reverse-KL descent signal $\delta_t = +u_t$ added to the GRPO advantage, a reward-anchored baseline rather than a direct implementation of OPSD [38]; see Section 4.1), and +AntiSD (Algorithm 1). The privileged context c concatenates an outcome-verified correct rollout from the same group (DAPO-Math-17k has final answers but no reference solutions) with a binary correctness-feedback string; when no rollout in the group is correct, the teacher receives the feedback without a solution trace (no dataset fallback). AntiSD's gate is auto-calibrated from the first 5 training steps (run at $\lambda = 0$): we set H_{warm} to the median of the five warm-up step-level values of H (so that one atypical batch does not determine it; this choice was not tuned) and $\tau_{\mathrm{down}} = 0.93\, H_{\mathrm{warm}}$, with the gate re-enabling once H recovers to H_{warm}. The 0.93 multiplier is hand-chosen; we use it unchanged across all five models without per-model retuning, but have not shown that it is optimal for any of them or that it transfers beyond them (Section 6). Held-out evaluation reports avg@32 on AIME 2024 [35] / 2025 [36] / 2026 [37] and HMMT 2025 [3], and avg@4 on MinervaMath [12]. Full settings and example teacher prompts are in Appendices B and C.

4.1 Main results

Table 1 reports avg@32 (avg@4 on Minerva) at each (model, method)'s best-Avg checkpoint. Three patterns hold: *(i) AntiSD reaches GRPO's accuracy in a fraction of the steps*, with a speedup of 2–10× across all five models. The largest speedups appear on the smaller models with weaker GRPO baselines (Qwen3-4B-IT-2507 10×, Olmo3-7B-IT 9.5×, Qwen3-8B 5×); the speedup shrinks but stays positive on the two strongest baselines (Olmo3-7B-TK 2×, where GRPO already sits at 64.1; Qwen3-30B-A3B 2.9×, the 30B mixture-of-experts model). This early ignition is consistent with the diagnosis in Section 3.1: the per-token reward $-\varphi(u_t)$ is informative from the first step, so credit-assignment does not have to wait for sparse trajectory-level reward to propagate through the policy. *(ii) AntiSD's final mean accuracy exceeds GRPO's on every model*, by $+2.1$ to $+11.5$ points (Avg). The gap is large on the weaker baselines ($+5.3$ to $+11.5$ on Qwen3-8B, Qwen3-4B-IT-2507, Olmo3-7B-IT), still substantial at scale ($+7.7$ on Qwen3-30B-A3B), and narrowest on the strongest GRPO baseline Olmo3-7B-TK ($+2.1$), where GRPO at 64.1 and the un-trained base at 62.0 leave little headroom on DAPO-Math-17k. Per-benchmark, 4 of 5 models win on every individual benchmark; the lone near-tie is a -0.6 pp gap on MinervaMath for Olmo3-7B-TK, ruling out the explanation that one easy benchmark inflates the mean. The gain is consistent with our prediction that biasing optimization toward deliberation tokens helps where GRPO's sparse signal provides little credit. *(iii) Default self-distillation underperforms the GRPO baseline* on every model, often by a wide margin (Qwen3-8B Avg: 30.6 vs 57.4). The mechanism behind this collapse, and the entropy dynamics that distinguish it from AntiSD, are examined in Section 4.2.

Training-seed variability. Avg@32 reduces evaluation-sampling noise but not training-seed variance, so we train three seeds on Qwen3-4B-IT-2507 (including the τ_{down}=0.90 ablation) and on Olmo3-7B-TK, the setting with the smallest Avg gap ($+2.1$). The three-seed means (Table 2) are close to the

Table 1: **Main results** (accuracy %). AIME24/25/26 and HMMT25: avg@32; Minerva: avg@4. Subscript on *Avg* = peak-mean step; *Speedup* = GRPO's best-Avg step / this row's first-reach step (×: never reached). **Bold** = column best within each model block; highlighted row is canonical AntiSD.

Method	AIME24	AIME25	AIME26	HMMT25	Minerva	Average	Speedup
Qwen3-8B	25.5	21.3	17.0	10.8	39.0	22.7	–
+GRPO	73.5	65.2	64.2	39.2	45.1	$57.4_{@200}$	1.0×
+SD	40.1	30.5	26.9	14.9	40.7	$30.6_{@200}$	×
+AntiSD	**78.4**	**73.4**	**73.7**	**54.4**	**48.5**	**$65.7_{@180}$**	**5.0×**
Qwen3-4B-IT-2507	62.1	48.2	53.8	30.4	43.4	47.6	–
+GRPO	67.8	57.7	63.5	34.1	33.2	$51.3_{@200}$	1.0×
+SD	59.8	45.8	52.0	28.8	43.0	$45.9_{@10}$	×
+AntiSD	**76.6**	**70.2**	**74.4**	**46.7**	**46.4**	**$62.8_{@100}$**	**10.0×**
Olmo3-7B-IT	53.3	39.7	44.4	25.9	38.7	40.4	–
+GRPO	57.0	45.3	52.1	31.2	29.1	$43.0_{@190}$	1.0×
+SD	54.5	41.8	46.6	24.4	38.5	$41.1_{@10}$	×
+AntiSD	**62.4**	**49.1**	**55.2**	**32.3**	**42.4**	**$48.3_{@200}$**	**9.5×**
Olmo3-7B-TK	74.6	68.7	73.3	48.2	45.2	62.0	–
+GRPO	76.5	71.7	75.3	50.5	**46.4**	$64.1_{@80}$	1.0×
+SD	77.2	68.5	74.0	48.3	44.8	$62.6_{@40}$	×
+AntiSD	**77.6**	**75.3**	**76.1**	**56.2**	45.8	**$66.2_{@40}$**	**2.0×**
Qwen3-30B-A3B	28.1	22.3	21.8	11.7	43.5	25.5	–
+GRPO	74.1	66.8	65.5	42.2	47.1	$59.1_{@150}$	1.0×
+SD	40.9	32.5	34.0	20.9	44.1	$34.5_{@60}$	×
+AntiSD	**79.4**	**75.6**	**74.1**	**55.2**	**49.7**	**$66.8_{@140}$**	**2.9×**

Table 2: **Training-seed variability** (avg@32): mean ± standard deviation over 3 training seeds (Appendix D.3).

Method	*Qwen3-4B-IT-2507* AIME25	*Qwen3-4B-IT-2507* HMMT25	*Olmo3-7B-TK* AIME25	*Olmo3-7B-TK* HMMT25
+GRPO	57.3 ± 1.1	34.1 ± 1.2	71.5 ± 0.7	50.3 ± 0.9
+AntiSD	69.9 ± 1.2	46.5 ± 1.3	75.1 ± 0.8	55.6 ± 1.1
+AntiSD (τ_{down}=0.90)	57.1 ± 1.6	40.3 ± 1.4	–	–

single-run values in Table 1; they keep AntiSD above GRPO on both benchmarks for both models and canonical AntiSD above τ_{down}=0.90. The remaining Table 1 settings are single-seed.

Matched OPSD baseline. Our +SD is a reward-anchored sampled-token baseline rather than a direct implementation of OPSD [38], so we also run OPSD's main objective on Qwen3-8B under the Table 1 protocol: a frozen initial-policy teacher, pointwise-clipped forward KL and full-vocabulary distillation, without a GRPO objective (Appendix D.6). It reaches 33.5 Avg, above +SD (30.6) but well below GRPO (57.4; Table 12). OPSD's original setting conditions the teacher on dataset reference solutions, and there OPSD is reported to outperform GRPO on math reasoning; DAPO-Math-17k has no reference solutions, so privileged traces come from correct peer rollouts and unsolved prompts receive feedback only. This run therefore tests OPSD's objective in our data setting rather than reproducing its reported results, and since several design axes differ, it does not isolate a single design choice. The result is consistent with reports of degradation or instability of standard self-distillation in challenging reasoning settings [9; 32; 13], including on the same Qwen3-8B/DAPO-Math-17k setting [9].

A natural concern is whether AntiSD trades sampled coverage for single-rollout accuracy. Figure 3 plots HMMT 2025 (the hardest benchmark) pass@k for $k \in \{1, 2, 4, 8, 16, 32\}$ at each row's HMMT25-peak checkpoint for all five models: AntiSD is at or above GRPO at every plotted k on every model and remains above it at $k = 32$ on all five, with a model-dependent gap that widens with k on some models and narrows on others. We read this as higher sampled outcome coverage within 32 attempts, not as a direct measure of semantic trajectory diversity or of problems GRPO cannot reach. On Olmo3-7B-IT, the smallest HMMT25

Table 3: Initial code probe (Qwen3-8B, avg@10). **Bold** marks the best per column.

Method	HumanEval+	MBPP+
+GRPO	40.4	61.0
+AntiSD	**41.6**	**63.3**

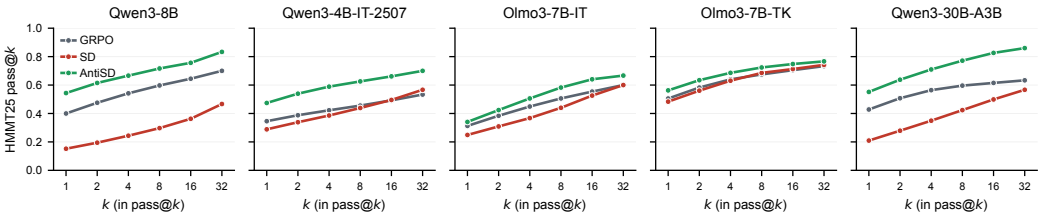


Figure 3: HMMT25 pass@k at each row's HMMT25-peak snapshot. AntiSD is at or above GRPO at every k on all five models; pass@k measures sampled outcome coverage, not trajectory diversity.

margin in Table 1, the added coverage does not come at the cost of modal-answer accuracy (maj@32: 43.3 for AntiSD vs 40.0 for GRPO; Appendix D.7).

Initial code probe. As preliminary evidence that the empirical benefit may extend beyond math, we run the same setup on the Dolci-RLZero code RL dataset [21] and evaluate on HumanEval+ and MBPP+ [16] (Table 3). This is a single-model, two-benchmark probe on Qwen3-8B: AntiSD improves over GRPO by $+1.2$ points on HumanEval+ and $+2.3$ on MBPP+. It does not establish parity with the math results or show that the token-level mechanism is domain-general.

4.2 Training dynamics

Figure 4 traces six training-time signals through the run. AntiSD ignites earliest: truncation-corrected train reward climbs to ~ 0.95 within ~ 30 steps on Qwen3-8B and Qwen3-4B-IT-2507, a regime GRPO reaches only after ~ 150 steps and SD never reaches, with HMMT25 and AIME26 moving in lockstep. The Qwen3-4B-IT-2507 plateau sits near 0.95 rather than 1.0 and drifts slightly late in training; held-out accuracy does not drop, so this is saturation against the DAPO-Math problem distribution – once almost every sampled problem is solved, the surviving gradient signal is noise – rather than overfitting.

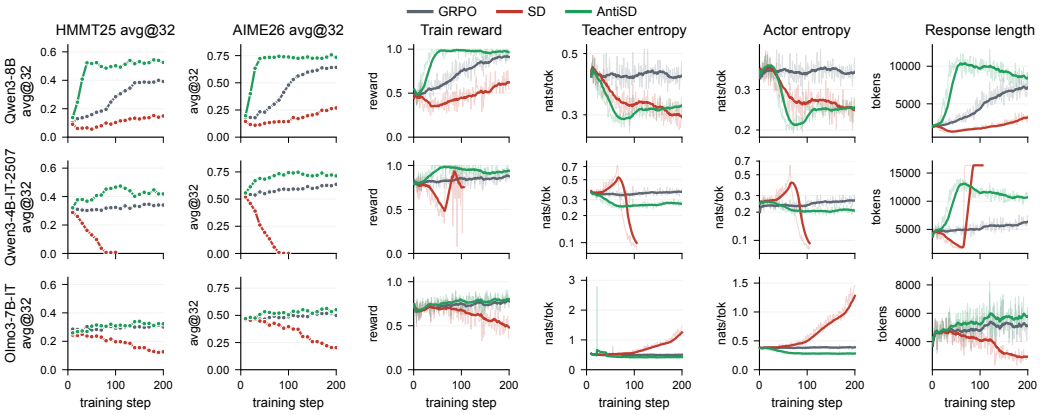


Figure 4: **Training dynamics** on three models (rows) along six axes (columns). Faded traces are raw values; bold traces are 20-step rolling means. Entropy panels use a logarithmic y-axis except in the Olmo3-7B-IT row (linear).

Default self-distillation diverges in opposite directions across model families. Both AntiSD and SD couple the student and teacher distributions, but their entropy traces tell different stories: AntiSD remains in a stable middle band on all three models, while SD's teacher and actor entropy collapse toward ~ 0.1 nats per token on Qwen3-4B-IT-2507 (over-confident on the shortcut answer template) and inflate past 1 nat per token on Olmo3-7B-IT (drift away from useful tokens). This is exactly the bidirectional failure mode that the sign reversal in Section 3.2 addresses; the same shortcut bias that explains SD's gap to GRPO in Table 1 is what is visibly amplifying or eroding teacher entropy here. Its sharpest expression is the Qwen3-4B-IT-2507 collapse around step 80: held-out accuracy to zero, response length pinned at the training response-length limit, and both entropies spiking, all within a single step window before the run terminates.

4.3 Ablations

AntiSD adds three components on top of the GRPO advantage: sign-reversed reward $-\varphi(u_t)$, the JSD/softplus shape, and an entropy-triggered gate. Sign reversal is the dominant lever and was already established in Table 1: removing it (default SD) drops Qwen3-8B Avg from 65.7 to 30.6. We focus the remaining ablations on the other two components and on the privileged context itself, reporting both training-curve health (does the run survive?) and held-out accuracy on Qwen3-4B-IT-2507, the most failure-prone model in our suite.

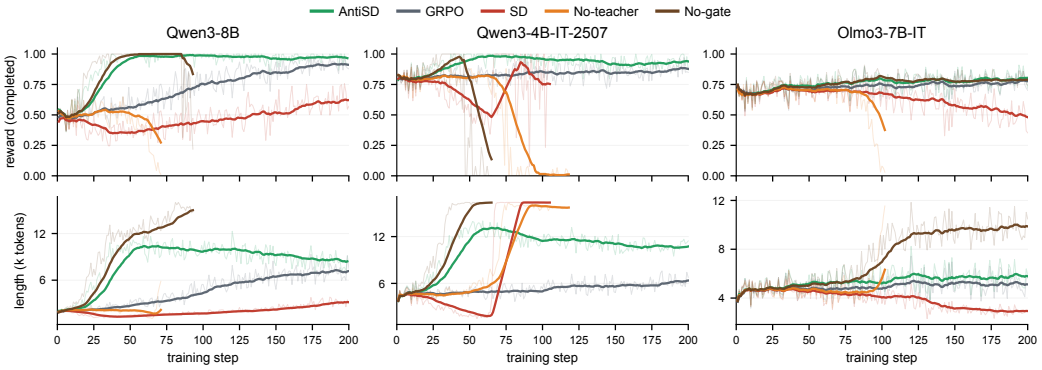


Figure 5: **Mechanism ablations: failure modes.** Top: reward over non-truncated rollouts. Bottom: mean response length. Line truncation indicates run termination after collapse.

No-teacher: self-reinforcement collapse. Removing the teacher entirely – so the per-token signal becomes a function of the student's log-probability alone, with no teacher–student differential – collapses on all three models within $\sim$ 70–100 training steps (Figure 5, orange). The resulting positive-feedback signal reinforces whatever the policy already emits – a textbook self-reinforcement collapse and the strongest evidence in our suite that AntiSD's gain depends on the privileged information identity from Section 3.1, not on a generic shaping of student log-probabilities. This contrasts with recent self-reward methods [39; 6], which keep an external signal (typically majority-vote agreement across rollouts) rather than removing all conditioning. AntiSD's privileged-context conditioning instead keeps pass@k rising with k and above GRPO out to $k = 32$ (Figure 3), which argues against collapse onto a single majority answer; part of its gain survives even an irrelevant privileged solution (Appendix D.5).

No-gate: stabilization is model-conditional. Removing the entropy gate (AntiSD term always on) is the most striking model-dependent failure (Figure 5, brown). On Qwen3-8B and Qwen3-4B-IT-2507 the No-gate run *ignites faster* than canonical AntiSD – reward reaches ~ 0.97 around step 40 – before collapsing near steps 90 and 50, respectively, as the teacher's per-token entropy approaches zero, at which point the per-token signal degenerates because the teacher has absorbed the answer template. On Olmo3-7B-IT the same configuration survives the full 200 steps and even attains the highest plateau on this model. The asymmetry tracks initial teacher entropy in the runs tested: Qwen models start at ≈ 0.4 nats per token (close to the absorption threshold) while Olmo starts higher, leaving headroom that the gate would have monitored. Whether the gate is needed *at all* is thus associated with initial teacher entropy; how much its *threshold value* matters is a separate, model-conditional question (below). We keep the gate in all runs as cross-model insurance.

Component sensitivity. Table 4 factorises the remaining design choices on Qwen3-4B-IT-2507 along three knobs: *Div* (JSD vs reverse-KL ascent), the gate deactivation threshold τ_{down}, and *Compose* (additive vs multiplicative shaping of the GRPO advantage). *Threshold sensitivity is model-conditional.* Loosening τ_{down} from 0.93 to 0.90 drops Avg by 8.3 points (62.8 $\rightarrow$ 54.5, single seed); the corresponding AIME25/HMMT25 drop persists across three training seeds (Table 2); on Qwen3-8B the same loosening leaves Avg essentially unchanged (65.9 vs canonical 65.7; Appendix D.1). The canonical 0.93 is the single hand-chosen value used for all five models, not a per-model optimum; threshold values were swept only on the two Qwen models. *Additive composition outperforms multiplicative.* Replacing additive with multiplicative drops Avg by 6.3 points (62.8 $\rightarrow$ 56.5) and halves the speedup (10$\times$ $\rightarrow$ 5$\times$): when A^{seq} is small, the multiplicative form scales the AntiSD term down toward zero, removing the deliberation push exactly when GRPO's ORM signal is

Table 4: **Component sensitivity on Qwen3-4B-IT-2507.** Single-knob deviations from canonical AntiSD (highlighted). *Speedup* as in Table 1. **Bold** = column best among +AntiSD rows.

Method	Div	τ_{down}	Compose	AIME24	AIME25	AIME26	HMMT25	Minerva	Average	Speedup
GRPO	–	–	–	67.8	57.7	63.5	34.1	33.2	$51.3_{@200}$	1.0×
+AntiSD	rev. KL	0.93	add.	64.1	49.0	58.9	32.0	43.7	$49.5_{@10}$	×
	JSD	none	add.	72.4	66.5	72.6	44.9	46.9	$60.6_{@30}$	**10.0**×
	JSD	0.90	add.	68.8	56.5	63.1	39.6	44.6	$54.5_{@20}$	**10.0**×
	JSD	0.95	add.	**77.6**	69.4	63.4	44.7	46.2	$60.3_{@110}$	6.7×
	JSD	0.93	mult.	70.8	61.5	67.6	36.6	46.1	$56.5_{@170}$	5.0×
	JSD	0.93	add.*	74.8	65.3	70.1	39.4	**51.8**	$60.3_{@20}{}^*$	**10.0**×
	JSD	0.93	add.	76.6	**70.2**	**74.4**	**46.7**	46.4	$\mathbf{62.8}_{@\mathbf{100}}$	**10.0**×

$*$ Gate statistic computed from the student instead of the teacher.

uninformative. The gate-off row's 60.6 peak at step 30 is a transient pre-collapse value (Figure 5, brown), not a sustainable plateau.

Mixing weight and context quality. In a single-seed $\lambda_{\max}$ sweep on Qwen3-4B-IT-2507 (Appendix D.4), peak Avg stays within 62.4–62.8 for $\lambda_{\max} \in [0.1, 0.75]$ (61.0 at 1.0; GRPO 51.3), while ignition speed varies (3.3× at 0.1, 10× at 0.25–0.75). In a single-seed run, degraded privileged contexts give lower Avg (Appendix D.5: 62.8 canonical, 61.9 dataset answer only, 59.6 incorrect peer rollout, 57.0 another problem's solution), yet all stay above GRPO (51.3).

4.4 Beyond GRPO saturation

A practical question is whether AntiSD must be trained from the base model, or whether the advantage shaping can be applied as a refinement on top of an existing GRPO checkpoint. We resume the Qwen3-8B GRPO run at step 200 – the saturation point in Table 1 – and continue with the canonical AntiSD configuration for 50 further steps. Optimizer state, dataloader index, and reference policy are inherited from the GRPO run via the standard verl [25] resume path; the gate threshold is recalibrated against the new H_{warm} to account for the shifted teacher-entropy distribution of the saturated policy.

Table 5: **Continual AntiSD on Qwen3-8B.** +*AntiSD*† resumes from GRPO@200. *Speedup* is defined only for runs trained from the base model; the continual run's cost is given by its *Steps* and *Avg* subscript, both counted post-resume. **Bold** = column best.

Method	Steps	AIME24	AIME25	AIME26	HMMT25	Minerva	Average	Speedup
GRPO	200	73.5	65.2	64.2	39.2	45.1	$57.4_{@200}$	1.0×
+AntiSD	200	**78.4**	**73.4**	**73.7**	**54.4**	48.5	$\mathbf{65.7}_{@\mathbf{180}}$	**5.0**×
+AntiSD†	+50	74.9	72.4	73.0	54.0	**50.6**	$65.0_{@30}$	–

Table 5 contrasts three policies: the saturated GRPO baseline at step 200, AntiSD trained from base, and continual AntiSD that initialises from the GRPO checkpoint. Continual AntiSD essentially matches the from-base peak (65.0 vs 65.7 Avg) after 30 post-resume steps on top of GRPO's 200, demonstrating that AntiSD's advantage stacks on top of GRPO rather than replacing it: the deliberation-token reward signal remains informative even at the saturation point that GRPO's trajectory-level reward cannot push past. The same experiment on Qwen3-4B-IT-2507 (Appendix D.2) comes within ~ 1 pp of the from-base peak before settling slightly lower, suggesting GRPO's basin admits most but not all of AntiSD's deliberation pressure.

5 Related Work

On-policy self-distillation. A series of recent works develops on-policy self-distillation along parallel axes [38; 7; 33; 22; 32], all using a teacher–student log-ratio in which the teacher is the student conditioned on privileged context (a verified solution and any environment feedback). These methods build on on-policy distillation [1; 4] (student-sampled rollouts with an external teacher) and learning under privileged information [26; 17], and share the same gradient direction. AntiSD inverts that direction, replacing descent on reverse KL with bounded ascent on Jensen–Shannon divergence; an auto-calibrated entropy-triggered gate then disables the term once the teacher's per-token entropy collapses. Concurrent work [8] also reverses the self-teacher signal on top of GRPO, reinforcing

tokens of successful rollouts where the student departs from the teacher's prediction; AntiSD reverses it on every rollout, motivated by the conditional-PMI polarity, with a bounded JSD shape and an entropy gate.

Diagnoses of self-distillation. Self-distillation has been diagnosed as degrading reasoning capability, directly in mathematical reasoning across several model families [9] and implicitly via its framing as a response-compression tool [22]; degradation or instability of standard self-distillation is also reported in multimodal reasoning [32] and under prolonged SDPO training [13]. Several self-distillation methods [7; 33; 32] report mainly on simpler benchmarks, not the AIME / HMMT-class problems where we observe the clearest failure; OPSD [38], which is reported to outperform GRPO on math reasoning, conditions its teacher on dataset reference solutions (Section 4.1). A broader line of on-policy distillation diagnostics [4; 14; 13; 30] documents teacher–student capability gaps and distribution mismatch without isolating self-distillation from external-teacher OPD. We confirm the symptom across 4B–30B models of two families (Section 4) but trace it to the per-token signal itself: under privileged context it is conditional PMI between the next token and that context (Section 3.1), which biases credit toward tokens the context already implies and away from deliberation tokens. AntiSD acts on this mechanism by reversing the per-token sign rather than by reweighting samples or filtering teacher confidence.

Process reward models and reward shaping. To address sparse credit assignment in RLVR, a separate line of work trains process reward models that score intermediate reasoning steps [15; 27; 19; 23; 11], either from human annotations or from Monte Carlo rollout estimates of step value, while implicit-reward methods such as PRIME [2] derive process rewards from preference signals jointly with the policy. The raw log-ratio u_t is the increment $\Phi_t - \Phi_{t-1}$ of the model's own log-posterior $\Phi_t = \log \pi_\theta(c \mid x, y_{\le t})$; it telescopes to the trajectory-level PMI $\log \pi_\theta(c \mid x, y) - \log \pi_\theta(c \mid x)$ (Lemma 3), a potential-based shaping in the sense of Ng et al. [20]. Canonical AntiSD applies the nonlinear $-\varphi(u_t)$ and the gate, so this invariance covers the raw signal, not the implemented objective. Like PRM-based training, AntiSD supplies a dense process-level signal; unlike it, the signal comes from the self-teacher without a separately trained reward model, step-level supervision or Monte Carlo rollouts, and it is not equivalent to a learned PRM. Our contribution is to identify the conditional PMI bias of this shaping signal (inflating shortcut tokens, suppressing deliberation tokens) and correct it through gradient-direction inversion rather than as an additional reward channel.

6 Limitations and Conclusion

Limitations. Our three-seed results cover only Qwen3-4B-IT-2507 and Olmo3-7B-TK on AIME25 and HMMT25. All other results are single-seed: the other Table 1 models and benchmark columns, the component ablations (except τ_{down}=0.90 on Qwen3-4B-IT-2507), the $\lambda_{\max}$ sweep, the context-quality ablation, and the matched OPSD, code-probe and continual runs. The 0.93 gate multiplier is hand-chosen and eliminating it remains an open problem; in a single-seed sweep, peak Avg varied little only over $\lambda_{\max} \in [0.1, 0.75]$, while ignition speed did vary. Gate guidance (Appendix E): whether the gate is needed at all tracks initial teacher entropy in the runs tested; its threshold value matters strongly on Qwen3-4B-IT-2507 but little for peak Avg on Qwen3-8B, and was swept only on these two models; and with another problem's solution as c, AntiSD stayed above GRPO in a single-seed run. AntiSD assumes a verifiable reward and a privileged context c (here, outcome-verified peer solutions plus correctness feedback), and its benefit depends on the quality of c; we expect it to help on verifiable tasks with such a context and headroom on the training distribution, and it does not directly apply to open-ended tasks without one. Our direct evidence for the self-distillation failure mode is limited to hard math, and our claims are scoped to math reasoning: code is an initial single-model probe, and broader code evaluation is the primary open item. Grounding, tool-mediated code execution, function calling and agentic settings are untested, and we do not test for gains in instruction following; IFEval and MMLU-Pro only check for regression after math-only training (no point-estimate regression vs GRPO; Appendix D.8), which is not evidence of equivalence. We do not include a matched comparison against PRM-trained policies, an important next step. We also do not include same-prompt trajectory comparisons between trained +SD and AntiSD checkpoints; our mechanism evidence is token-level and aggregate.

Conclusion. Through conditional PMI with the privileged context, default on-policy self-distillation rewards tokens the context already implies and penalises deliberation. AntiSD reverses this signal with bounded JSD ascent under an entropy gate, reaching GRPO's math accuracy in 2 to $10\times$ fewer steps and exceeding it by up to 11.5 points across five models.

Acknowledgments and Disclosure of Funding

We thank the anonymous reviewers and the area chair for their constructive feedback, which improved this paper. **Funding:** this work was supported by the authors' affiliated institutions, Xiaohongshu Inc. and the Institute of Automation, Chinese Academy of Sciences. **Competing interests:** the authors declare no competing interests.

References

[1] Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The twelfth international conference on learning representations*, 2024.

[2] Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Yuchen Zhang, Jiacheng Chen, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, et al. Process reinforcement through implicit rewards. *arXiv preprint arXiv:2502.01456*, 2025.

[3] Jasper Dekoninck, Nikola Jovanović, Tim Gehrunger, Kári Rögnvalddson, Ivo Petrov, Chenhao Sun, and Martin Vechev. Beyond benchmarks: MathArena as an evaluation platform for mathematics with LLMs. *arXiv preprint arXiv:2605.00674*, 2026.

[4] Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. *arXiv preprint arXiv:2603.25562*, 2026.

[5] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[6] Bingxiang He, Yuxin Zuo, Zeyuan Liu, Shangziqi Zhao, Zixuan Fu, Junlin Yang, Cheng Qian, Kaiyan Zhang, Yuchen Fan, Ganqu Cui, et al. How far can unsupervised RLVR scale LLM training? *arXiv preprint arXiv:2603.08660*, 2026.

[7] Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, et al. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.

[8] Jeonghye Kim, Jiwon Jeon, Dongsheng Li, and Yuqing Yang. Rebellious student: Reversing teacher signals for reasoning exploration with self-distilled RLVR. *arXiv preprint arXiv:2605.10781*, 2026.

[9] Jeonghye Kim, Xufang Luo, Minbeom Kim, Sangmook Lee, Dohyung Kim, Jiwon Jeon, Dongsheng Li, and Yuqing Yang. Why does self-distillation (sometimes) degrade the reasoning capability of LLMs? *arXiv preprint arXiv:2603.24472*, 2026.

[10] Kimi Team, Angang Du, Bofei Gao, Bowen Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1.5: Scaling reinforcement learning with LLMs. *arXiv preprint arXiv:2501.12599*, 2025.

[11] Nakyung Lee, Sangwoo Hong, and Jungwoo Lee. Efficient process reward modeling via contrastive mutual information. *arXiv preprint arXiv:2604.10660*, 2026.

[12] Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35:3843–3857, 2022.

[13] Gengsheng Li, Tianyu Yang, Junfeng Fang, Mingyang Song, Mao Zheng, Haiyun Guo, Dan Zhang, Jinqiao Wang, and Tat-Seng Chua. Unifying group-relative and self-distillation policy optimization via sample routing. *arXiv preprint arXiv:2604.02288*, 2026.

[14] Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huan-ang Gao, Wenkai Yang, Zhiyuan Liu, et al. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026.

[15] Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The twelfth international conference on learning representations*, 2023.

[16] Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by ChatGPT really correct? Rigorous evaluation of large language models for code generation. *Advances in neural information processing systems*, 36:21558–21572, 2023.

[17] David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. *arXiv preprint arXiv:1511.03643*, 2015.

[18] Kevin Lu and Thinking Machines Lab. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20251026. https://thinkingmachines.ai/blog/on-policy-distillation.

[19] Liangchen Luo, Yinxiao Liu, Rosanne Liu, Samrat Phatale, Meiqi Guo, Harsh Lara, Yunxuan Li, Lei Shu, Yun Zhu, Lei Meng, et al. Improve mathematical reasoning in language models by automated process supervision. *arXiv preprint arXiv:2406.06592*, 2024.

[20] Andrew Y Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 99, pages 278–287. Citeseer, 1999.

[21] Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3, 2025. URL `https://arxiv.org/abs/2512.13961`.

[22] Hejian Sang, Yuanda Xu, Zhengze Zhou, Ran He, Zhipeng Wang, and Jiachen Sun. On-policy self-distillation for reasoning compression. *arXiv preprint arXiv:2603.05433v1*, 2026.

[23] Amrith Setlur, Chirag Nagpal, Adam Fisch, Xinyang Geng, Jacob Eisenstein, Rishabh Agarwal, Alekh Agarwal, Jonathan Berant, and Aviral Kumar. Rewarding progress: Scaling automated process verifiers for LLM reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=A6Y7AqlzLW`.

[24] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[25] Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. HybridFlow: A flexible and efficient RLHF framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pages 1279–1297, 2025.

[26] Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural networks*, 22(5-6):544–557, 2009.

[27] Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-Shepherd: Verify and reinforce LLMs step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9426–9439, 2024.

[28] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems (Datasets and Benchmarks Track)*, 2024.

[29] Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, et al. MiMo-V2-Flash technical report. *arXiv preprint arXiv:2601.02780*, 2026.

[30] Yuanda Xu, Hejian Sang, Zhengze Zhou, Ran He, and Zhipeng Wang. PACED: Distillation and on-policy self-distillation at the frontier of student competence. *arXiv preprint arXiv:2603.11178*, 2026.

[31] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[32] Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled RLVR. *arXiv preprint arXiv:2604.03128*, 2026.

[33] Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models. *arXiv preprint arXiv:2602.12275*, 2026.

[34] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

[35] Yifan Zhang and Team Math-AI. American Invitational Mathematics Examination (AIME) 2024, 2024.

[36] Yifan Zhang and Team Math-AI. American Invitational Mathematics Examination (AIME) 2025, 2025.

[37] Yifan Zhang and Team Math-AI. American Invitational Mathematics Examination (AIME) 2026, 2026.

[38] Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

[39] Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. Learning to reason without external rewards. *arXiv preprint arXiv:2505.19590*, 2025.

[40] Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

Anti-Self-Distillation for Reasoning RL via Pointwise Mutual Information Supplementary Material

Table of Contents

A Proofs and Deferred Statements . . . 14
B Hyperparameters . . . 16
C Self-Teacher Context Examples . . . 17
D Additional Experiments . . . 18
D.1 Component sensitivity on Qwen3-8B . . . 18
D.2 Continual AntiSD on Qwen3-4B-IT-2507 . . . 19
D.3 Multi-seed results . . . 19
D.4 Sensitivity to $\lambda_{\max}$ . . . 19
D.5 Privileged-context quality . . . 20
D.6 Matched OPSD baseline: details . . . 20
D.7 Sampled outcome coverage and modal-answer accuracy . . . 21
D.8 Capability regression after math-only training . . . 21
E Limitations and Broader Impacts . . . 22

A Proofs and Deferred Statements

This appendix collects the derivations summarised in Section 3: the reverse-KL gradient identity (Equation (3), Lemma 1); the PMI characterization of u_t under self-distillation (Equation (4), Lemma 2) and its trajectory-level telescope into a potential-based shaping term (Lemma 3); the JSD f-divergence shape used in Equation (6) (Lemma 4); the properties of φ relied on in Section 3.2 (Lemma 5); the polarity of descent on a general f-divergence (Lemma 6); and the total-variation identity behind observation (O2) (Remark 7). All distributions are over the next-token vocabulary; we suppress the conditioning $(x, y_{<t})$ where it is fixed.

Lemma 1 (Reverse-KL gradient identity, Equation (3))**.** *Let $\pi_S(\cdot) = \pi_\theta(\cdot \mid x, y_{<t})$ and $\pi_T(\cdot) = \pi_\theta(\cdot \mid x, c, y_{<t})$ with stop-gradient on π_T, and write $u_v := \log \pi_T(v) - \log \pi_S(v)$. Then*

$$\nabla_\theta \, D_{\mathrm{KL}}(\pi_S \| \pi_T) \;=\; -\,\mathbb{E}_{v\sim\pi_S}[u_v \cdot \nabla_\theta \log \pi_S(v)] \,.$$

Proof. Expanding $D := D_{\mathrm{KL}}(\pi_S\|\pi_T) = \sum_v \pi_S(v)\,(\log \pi_S(v) - \log \pi_T(v))$ and applying the product rule,

$$\nabla_\theta D \;=\; \underbrace{\sum_v \nabla_\theta \pi_S(v)\,(\log \pi_S(v) - \log \pi_T(v))}_{\text{(I)}} \;+\; \underbrace{\sum_v \pi_S(v)\,\nabla_\theta(\log \pi_S(v) - \log \pi_T(v))}_{\text{(II)}} .$$

Term (II) vanishes: $\nabla_\theta \log \pi_T(v) = 0$ by stop-gradient, and $\sum_v \pi_S(v)\,\nabla_\theta \log \pi_S(v) = \sum_v \nabla_\theta \pi_S(v) = \nabla_\theta \sum_v \pi_S(v) = \nabla_\theta 1 = 0$. For term (I), the score-function identity $\nabla_\theta \pi_S(v) = \pi_S(v)\,\nabla_\theta \log \pi_S(v)$ gives

$$\text{(I)} \;=\; \mathbb{E}_{v\sim\pi_S}[(\log \pi_S(v) - \log \pi_T(v))\,\nabla_\theta \log \pi_S(v)] \;=\; -\,\mathbb{E}_{v\sim\pi_S}[u_v \nabla_\theta \log \pi_S(v)] \,,$$

where $\log \pi_S - \log \pi_T = -u$. Combining (I) + (II) proves the claim. □

Lemma 2 (PMI characterization, Equation (4))**.** *Under the self-distillation parameter sharing* $\pi_S(\cdot) = \pi_\theta(\cdot \mid x, y_{<t})$ *and* $\pi_T(\cdot) = \pi_\theta(\cdot \mid x, c, y_{<t})$*,*

$$u_t \;=\; \log \frac{\pi_\theta(y_t \mid x, c, y_{<t})}{\pi_\theta(y_t \mid x, y_{<t})} \;=\; \mathrm{PMI}(y_t\,;\, c \mid x, y_{<t}).$$

Proof. Bayes' rule applied to the joint of (y_t, c) given $(x, y_{<t})$ gives

$$\pi_\theta(y_t \mid x, c, y_{<t}) \;=\; \frac{\pi_\theta(c \mid x, y_{\le t})\,\pi_\theta(y_t \mid x, y_{<t})}{\pi_\theta(c \mid x, y_{<t})},$$

so $\frac{\pi_\theta(y_t|x,c,y_{<t})}{\pi_\theta(y_t|x,y_{<t})} = \frac{\pi_\theta(c|x,y_{\le t})}{\pi_\theta(c|x,y_{<t})}$, and taking logs yields the conditional pointwise mutual information $\mathrm{PMI}(y_t; c \mid x, y_{<t})$. □

Lemma 3 (Trajectory-level potential shaping)**.** *Summing* u_t *over a complete trajectory telescopes to a sequence-level pointwise mutual information:*

$$\sum_{t=1}^{T} u_t \;=\; \log \pi_\theta(c \mid x, y) - \log \pi_\theta(c \mid x) \;=\; \mathrm{PMI}(y\,;\, c \mid x).$$

Hence the per-token contributions $\{u_t\}$ *are the increments of a potential* $\Phi_t := \log \pi_\theta(c \mid x, y_{\le t})$*, and for* $\delta_t = \pm u_t$ *(not for the nonlinear* $-\varphi(u_t)$ *of canonical AntiSD) the augmented advantage in Equation* (2) *is a potential-based reward shaping in the sense of Ng et al. [20]: it leaves the set of optimal policies invariant for any underlying scalar reward.*

Proof. By Lemma 2, $u_t = \log \pi_\theta(c \mid x, y_{\le t}) - \log \pi_\theta(c \mid x, y_{<t}) = \Phi_t - \Phi_{t-1}$. The sum telescopes to $\Phi_T - \Phi_0 = \log \pi_\theta(c \mid x, y) - \log \pi_\theta(c \mid x)$. The potential-based shaping invariance result follows directly [20]. □

Lemma 4 (JSD f-divergence shape, Equation (6))**.** *Write the symmetric Jensen–Shannon divergence in f-divergence form* $D_{\mathrm{JSD}}(\pi_S \| \pi_T) = \mathbb{E}_{\pi_T}[f(\pi_S/\pi_T)]$ *with generator* $f(r) = \frac{1}{2} r \log \frac{2r}{1+r} + \frac{1}{2} \log \frac{2}{1+r}$*. Then* $f'(r) = \frac{1}{2} \log \frac{2r}{1+r}$ *and, for* $r = \pi_S/\pi_T = e^{-u}$*,*

$$f'\!\left(\tfrac{\pi_S}{\pi_T}\right) \;=\; \tfrac{1}{2}(\log 2 - \mathrm{softplus}(u)) \;=\; -\varphi(u),$$

recovering the AntiSD advantage $A_t^{\mathrm{AntiSD}} = -\varphi(u_t)$ *via the score-function identity in Equation* (5)*.*

Proof. Differentiating f term by term, with $g(r) := \log \frac{2r}{1+r} = \log 2 + \log r - \log(1+r)$ and $g'(r) = \frac{1}{r} - \frac{1}{1+r} = \frac{1}{r(1+r)}$, the product rule gives

$$f'(r) = \tfrac{1}{2} g(r) + \tfrac{1}{2} r\, g'(r) - \tfrac{1}{2} \cdot \tfrac{1}{1+r} = \tfrac{1}{2}\Big[g(r) + \tfrac{1}{1+r} - \tfrac{1}{1+r}\Big] = \tfrac{1}{2} \log \tfrac{2r}{1+r}.$$

Substituting $r = e^{-u}$ gives $\frac{2r}{1+r} = \frac{2}{e^u+1}$, so $f'(e^{-u}) = \frac{1}{2}(\log 2 - \log(1+e^u)) = \frac{1}{2}(\log 2 - \mathrm{softplus}(u)) = -\varphi(u)$. □

Lemma 5 (Properties of φ)**.** *The shape* $\varphi(u) := \frac{1}{2}(\mathrm{softplus}(u) - \log 2)$ *from Equation* (6) *satisfies:*

(i) Strict monotonicity: $\varphi'(u) = \frac{1}{2}\sigma(u) > 0$*, where* $\sigma(u) = (1 + e^{-u})^{-1}$ *is the logistic function.*

(ii) Sign preservation: $\varphi(0) = 0$ *and* $\mathrm{sign}(\varphi(u)) = \mathrm{sign}(u)$.

(iii) One-sided bound: $\varphi(u) \ge -\frac{1}{2}\log 2$ *for all* $u \in \mathbb{R}$*, with equality attained as* $u \to -\infty$*;* φ *is unbounded above.*

Proof. (i) $\frac{d}{du}\mathrm{softplus}(u) = \frac{e^u}{1+e^u} = \sigma(u)$, so $\varphi'(u) = \frac{1}{2}\sigma(u) > 0$ for all u. (ii) $\mathrm{softplus}(0) = \log 2$, so $\varphi(0) = 0$; combined with (i), φ is strictly increasing through 0. (iii) $\mathrm{softplus}(u) = \log(1+e^u) > 0$ for all u, with $\inf_u \mathrm{softplus}(u) = 0$ attained as $u \to -\infty$, hence $\varphi(u) \ge -\frac{1}{2}\log 2$ with equality in the limit. As $u \to \infty$, $\mathrm{softplus}(u) \sim u \to \infty$, so φ has no upper bound. □

Properties (i)–(ii) ensure the shape inherits the per-token sign structure of Equations (3)–(4): $-\varphi$ flips that sign at the source, so deliberation tokens ($u_t < 0$) receive positive advantage and shortcut tokens ($u_t > 0$) receive negative advantage. Property (iii) bounds the AntiSD advantage at $\frac{1}{2}\log 2$ on the deliberation side – the side that observation (O2) flagged as both over-sampled and heavy-tailed – so the cap re-balances per-token gradient contributions against the lighter shortcut side. The entropy gate then disables the term entirely once u_t degenerates into floor-level noise.

The sign structure is not specific to the two divergences above. The next lemma shows that descent on any convex f-divergence inherits the shortcut/deliberation polarity of reverse KL (Section 3.1); the generator f only shapes its magnitude.

Lemma 6 (Polarity of f-divergence descent)**.** *Let $D_f(\pi_S\|\pi_T) = \mathbb{E}_{\pi_T}[f(r)]$ with $r = \pi_S/\pi_T = e^{-u}$ for a convex, differentiable generator f, with stop-gradient on π_T. Without loss of generality $f'(1) = 0$. Then descent on D_f has per-token advantage*

$$\delta_f(u) \;=\; -f'\big(e^{-u}\big)\,,$$

which is non-decreasing in u with $\delta_f(0) = 0$. Hence $\delta_f(u) \geq 0$ for $u > 0$ and $\delta_f(u) \leq 0$ for $u < 0$, with strict inequalities when f is strictly convex.

Proof. Replacing $f(r)$ by $f(r) - a(r-1)$ leaves D_f unchanged, since $\mathbb{E}_{\pi_T}[r-1] = \sum_v \pi_S(v) - \sum_v \pi_T(v) = 0$. The shift changes the per-token advantage below by the constant a, a baseline that leaves the expected gradient unchanged because $\mathbb{E}_{v\sim\pi_S}[\nabla_\theta \log \pi_S(v)] = 0$; we fix it by choosing $a = f'(1)$, i.e., $f'(1) = 0$, so that the advantage vanishes when $\pi_S = \pi_T$. The polarity statement refers to this normalisation. The score-function calculation of Equation (5) applies to any differentiable f: $\nabla_\theta D_f = \sum_v \pi_T(v)\, f'(r_v)\, \nabla_\theta \pi_S(v)/\pi_T(v) = \mathbb{E}_{v\sim\pi_S}[f'(r_v)\, \nabla_\theta \log \pi_S(v)]$, so descent ($-\nabla_\theta D_f$) has per-token advantage $-f'(e^{-u})$. By convexity f' is non-decreasing, and e^{-u} is strictly decreasing in u, so $-f'(e^{-u})$ is non-decreasing in u (strictly increasing if f is strictly convex); and $\delta_f(0) = -f'(1) = 0$. □

As sanity checks, reverse KL ($f(r) = r\log r$, normalised to $f'(r) = \log r$) gives $\delta_f(u) = u$, recovering Equation (3), and the JSD generator of Lemma 4 gives $\delta_f(u) = \varphi(u)$, whose sign reversal is the AntiSD advantage.

Remark 7 (Over-sampling in observation (O2))**.** *At a fixed context, let $V^- = \{v : \pi_S(v) > \pi_T(v)\}$, i.e., exactly the tokens with $u_v < 0$. Since $\sum_v(\pi_S(v) - \pi_T(v)) = 0$,*

$$\pi_S(V^-) - \pi_T(V^-) \;=\; \sum_{v\in V^-} \big(\pi_S(v) - \pi_T(v)\big) \;=\; \tfrac{1}{2}\sum_v \big|\pi_S(v) - \pi_T(v)\big| \;=\; \mathrm{TV}(\pi_S, \pi_T) \;\geq\; 0.$$

A token drawn from π_S therefore falls in V^- more often than a token drawn from π_T at the same context, by exactly the total-variation distance. The identity holds at each context and hence after averaging over a batch; it does not imply that V^- tokens form a majority of the batch, and the heavier negative lobe in Figure 2(b) is a separate empirical observation. Because $u_v = \log(\pi_T(v)/\pi_S(v))$ has no finite lower bound, a token sampled with non-negligible π_S but near-zero π_T yields a very negative u_v. Under an unbounded shape such as reverse-KL ascent (advantage $-u_v$), such values would give very large positive advantages; the AntiSD advantage is bounded on this side, $-\varphi(u) \leq \frac{1}{2}\log 2$ (Lemma 5(iii)).

B Hyperparameters

We evaluate five language models: **Qwen3-8B**, **Qwen3-4B-Instruct-2507** (Qwen3-4B-IT-2507), **Olmo-3-7B-Instruct** (Olmo3-7B-IT), **Olmo-3-7B-Think** (Olmo3-7B-TK), and **Qwen3-30B-A3B**. All models share the configuration below, with training at 32K maximum sequence length; the only model-conditional knob is the evaluation-time maximum sequence length, which is doubled for the thinking-model variant (Olmo-3-7B-Think) to accommodate longer chains-of-thought. AntiSD adds the gate parameters in the bottom block; the auto-calibration of H_{warm} and τ_{down} is described in Section 4 (Setup).

Algorithm 1 Anti-Self-Distillation (AntiSD) – one training step.

Require: Policy π_θ, batch $\{(x_i, y_i, c_i)\}_{i=1}^{B}$ with sequence-level GRPO advantage A_i^{seq}; hyperparameter $\lambda_{\max}$; gate state g and calibrated threshold τ_{down} (warmup baseline H_{warm}).

for each rollout i and token $t \in \{1, \ldots, T_i\}$ **do**
  $s_{i,t} \leftarrow \log \pi_\theta(y_{i,t} \mid x_i, y_{i,<t})$ ▷ student log-prob
  $t_{i,t} \leftarrow \text{stopgrad}(\log \pi_\theta(y_{i,t} \mid x_i, c_i, y_{i,<t}))$ ▷ teacher log-prob
  $u_{i,t} \leftarrow t_{i,t} - s_{i,t}$ ▷ $= \text{PMI}(y_{i,t};\, c_i \mid x_i, y_{i,<t})$, see Equation (4)
  $\varphi_{i,t} \leftarrow \frac{1}{2}(\text{softplus}(u_{i,t}) - \log 2)$ ▷ JSD f-divergence advantage; see Equation (6)
end for
$H \leftarrow \text{median}_{i,t}\, H[\pi_T(\cdot \mid x_i, y_{i,<t})]$ ▷ teacher entropy
Update gate: $g \leftarrow 1$ if $H \geq H_{\text{warm}}$, $g \leftarrow 0$ if $H < \tau_{\text{down}}$, else unchanged.
$\lambda \leftarrow g \cdot \lambda_{\max}$
$A_{i,t} \leftarrow A_i^{\text{seq}} - \lambda \cdot \text{stopgrad}(\varphi_{i,t})$ ▷ ascent on $D_{\text{JSD}}(\pi_S \| \pi_T)$; advantage is treated as a constant weight (cf. Eq. (5))
Update θ via standard policy gradient using $\{A_{i,t}\}$.

Table 6: Training and evaluation hyperparameters. AntiSD shares the GRPO/SD configuration and adds only the gate parameters in the bottom block.

Hyperparameter	**Value**
Optimizer	AdamW
Learning rate	1×10^{-6}
Training steps	200
Batch size (problems)	32
Rollouts per problem (group size)	8
Max sequence length (training)	32K (all models)
Max sequence length (evaluation)	32K (Qwen, Olmo3-7B-IT), 64K (Olmo3-7B-TK)
GRPO clip ratio	0.2
KL penalty coefficient	0 (no reference-policy KL term, following [34])
Reference model π_{ref}	frozen base model
Training rollout sampling	temperature 1.0, top-$p = 1.0$
Evaluation rollout sampling	temperature 0.7, top-$p = 0.95$
Evaluation rollouts per problem	32 (AIME, HMMT) / 4 (MinervaMath)
Hardware	8 NVIDIA H20 GPUs per node
GPUs per run	16 (4B–8B models), 32 (Qwen3-30B-A3B)
Wall-clock per 200-step run	10–75 h (depending on model and sequence length)
Framework	verl [25]
AntiSD mixing weight $\lambda_{\max}$	0.5
AntiSD warmup length W	5 steps
Warm-up baseline H_{warm}	median of the W step-level values of H
AntiSD deactivation threshold τ_{down}	$0.93 \cdot H_{\text{warm}}$
AntiSD reactivation threshold	H_{warm}

C Self-Teacher Context Examples

We show the context that the self-teacher $\pi_T(\cdot \mid x, y_{<t}, c)$ sees when re-evaluating the student's response. The teacher's input concatenates the original prompt, an outcome-verified correct rollout from the same group when one exists (otherwise this block is omitted and the teacher sees only the feedback string; the dataset is not used as a fallback), and a feedback string indicating correctness, followed by an instruction to re-solve the problem. The student's original response y is placed in the assistant role; the teacher then re-evaluates y's log-probabilities under this enriched context. Templates follow [7; 38] and are identical for math and code; only the verified-solution and feedback strings differ by task. Colors: original prompt, feedback string (a sub-component of c), student response y (re-evaluated by teacher).

Reading the template. The block under *Your previous attempt:* carries the verified solution (an outcome-verified peer rollout); the *Previous assessment:* line is the correctness feedback for the student's actual rollout y that follows in the assistant turn, not for the verified solution shown above.

The two slots therefore play complementary roles: the verified solution narrows the teacher's posterior toward the correct answer, while the assessment string indicates that the trajectory the teacher will now re-evaluate is the student's own (which may be wrong). The deliberate asymmetry between "correct reference" and "incorrect attempt" gives the teacher a contrastive signal, and matches the prompt structure used in prior on-policy self-distillation work [7; 38].

Math (AIME / HMMT / MinervaMath) — self-teacher context

[User] Solve the following math problem. Place the final answer in \boxed{}.
Find the number of integer pairs (a, b) with $1 \leq a, b \leq 100$ such that $\gcd(a, b) + \mathrm{lcm}(a, b) = a + b + 50$.

Your previous attempt:
(an outcome-verified correct rollout from the same group; omitted when no rollout in the group is correct)
Let $d = \gcd(a, b)$, $a = dm$, $b = dn$ ... so $d(1 + mn) = dm + dn + 50$ ...
\boxed{25}

Previous assessment: Your answer is incorrect.

Now solve this problem step by step.

[Assistant] *(student's original response y, re-evaluated by teacher)*
We start by writing $a = dm$, $b = dn$ with $\gcd(m, n) = 1$ ...
\boxed{30}

Code (Dolci-RLZero) — self-teacher context

[User] You are a coding expert. Write a correct Python function that solves the following problem; place the final code in a \texttt{```python} block.
Given a list of integers nums, partition it into the minimum number of strictly increasing subsequences and return that number.

Your previous attempt:
(an outcome-verified successful rollout from the same group; omitted when none exists)
```
from bisect import bisect_left
def partitions(nums): tails = []; for x in nums: ...; return len(tails)
```

Previous assessment: Your code passes 7 of 12 test cases.

Now solve this problem step by step.

[Assistant] *(student's original response y, re-evaluated by teacher)*
We track the tail of each subsequence in sorted order...
```
def partitions(nums): ...
```

The feedback string is the only task-conditional piece. For math we use a binary form ("Your answer is correct." / "Your answer is incorrect."); for code, we use a continuous form ("Your code passes N of M test cases.") matching the per-test fraction returned by the reward function. Both forms parallel the task's underlying score: math is exact-match boolean, while code's score is a fraction over executed test cases.

D Additional Experiments

This section gathers experimental results referenced from the main paper but deferred to the appendix for space. Each subsection corresponds to one of the experimental probes whose narrative is summarised in Section 4.

D.1 Component sensitivity on Qwen3-8B

Table 7 mirrors the Qwen3-4B-IT-2507 ablation in Section 4.3 on Qwen3-8B. Two patterns from the main analysis carry over: *rev. KL ascent collapses* (30.6 Avg, -35.1 pp from canonical) and *the gate is necessary on this model family* (the no-gate run shows a transient peak at step ~ 40 before collapsing by step ~ 90, echoing the dynamics in Section 4.3). The threshold-sensitivity story differs in direction: loosening from 0.93 to 0.90 slightly *improves* Avg on Qwen3-8B ($65.7 \rightarrow 65.9$), in stark contrast to the -8.3 pp drop on Qwen3-4B-IT-2507. Tightening to 0.95 slightly drops the peak ($65.7 \rightarrow 65.4$) and slows ignition by $\sim 4\times$. The canonical 0.93 is therefore not the per-model optimum on Qwen3-8B; it is the single hand-chosen value used for all five models without per-model retuning, and threshold values were swept only on the two Qwen models.

Table 7: **Component sensitivity on Qwen3-8B.** Same format as Table 4. *Speedup* uses GRPO's best-Avg step (200). **Bold** = column best among +AntiSD rows.

Method	Div	τ_{down}	Compose	AIME24	AIME25	AIME26	HMMT25	Minerva	Average	Speedup
GRPO	–	–	–	73.5	65.2	64.2	39.2	45.1	$57.4_{@200}$	1.0×
+AntiSD	rev. KL	0.93	add.	40.1	30.5	26.9	14.9	40.7	$30.6_{@200}$	×
	JSD	none	add.	75.5	68.7	69.2	45.9	48.7	$61.6_{@40}$†	**6.7**×
	JSD	0.90	add.	77.8	73.2	73.2	**57.0**	48.2	$\mathbf{65.9}_{@100}$	**6.7**×
	JSD	0.95	add.	76.3	**73.4**	**75.8**	51.9	49.7	$65.4_{@180}$	1.4×
	JSD	0.93	mult.	73.1	60.6	61.7	38.5	44.6	$55.7_{@130}$	×
	JSD	0.93	add.*	78.1	72.7	73.2	52.3	**50.9**	$65.4_{@40}$	**6.7**×
	JSD	0.93	add.	**78.4**	**73.4**	73.7	54.4	48.5	$65.7_{@180}$	5.0×

† The no-gate run on Qwen3-8B collapses by step $\sim$ 90 (cf. Section 4.3); the reported peak is from 8 pre-collapse checkpoints. $*$ Gate statistic computed from the student instead of the teacher.

D.2 Continual AntiSD on Qwen3-4B-IT-2507

We repeat the continual experiment from Section 4.4 on Qwen3-4B-IT-2507. Unlike Qwen3-8B, where continual AntiSD essentially closes the gap to from-base AntiSD (65.0 vs 65.7 Avg), continual on the smaller model peaks at 61.9 briefly at step $+20$ before drifting to a plateau of ≈ 60.5; the plateau is 2.3 pp short of the from-base peak (62.8).

Table 8: **Continual AntiSD on Qwen3-4B-IT-2507.** Same setup as Table 5 on the smaller model. *Speedup* is defined only for runs trained from the base model; the continual run's cost is given by its *Steps* and *Avg* subscript, both counted post-resume. **Bold** = column best.

Method	Steps	AIME24	AIME25	AIME26	HMMT25	Minerva	Average	Speedup
GRPO	200	67.8	57.7	63.5	34.1	33.2	$51.3_{@200}$	1.0×
+AntiSD	200	**76.6**	**70.2**	**74.4**	**46.7**	46.4	$\mathbf{62.8}_{@100}$	**10.0**×
+AntiSD†	+50	76.2	69.7	73.3	43.2	**47.0**	$61.9_{@20}$	–

The plateau is consistent with two interpretations: (i) GRPO's basin admits some, but not all, of the deliberation pressure that AntiSD applies from base; (ii) the auto-recalibrated gate threshold derived from the saturated policy's H_{warm} is mildly conservative on this model, leaving residual AntiSD gain unrealised. We do not attempt to disentangle these here; the practical takeaway is that continual AntiSD provides a strong fraction of the from-base improvement at a fraction of the cost, with a model-conditional ceiling.

D.3 Multi-seed results

Table 9 gives the three-seed results summarised in Table 2 next to the single-run values reported elsewhere in the paper. Variability is across 3 training seeds and is reported as mean $\pm$ standard deviation; evaluation is unchanged (avg@32). Qwen3-4B-IT-2507 is the model with the τ_{down} ablation, and Olmo3-7B-TK is the setting with the smallest Avg gap in Table 1 ($+2.1$). The three-seed means are close to the single-run values; they keep AntiSD above GRPO on both benchmarks for both models and canonical AntiSD above τ_{down}=0.90, whereas the AIME25 means of GRPO and τ_{down}=0.90 (57.3 vs 57.1) are not separated by the seeds. All other settings in the paper are single-seed, and we make no formal significance claims.

D.4 Sensitivity to $\lambda_{\max}$

Table 10 sweeps the AntiSD mixing weight $\lambda_{\max}$ on Qwen3-4B-IT-2507 with all other settings canonical (single seed; *Speedup* as in Table 1). The unreplicated peak-Avg point estimates remain between 62.4 and 62.8 over $\lambda_{\max} \in [0.1, 0.75]$, compared with 61.0 at $\lambda_{\max} = 1.0$ and GRPO's 51.3. $\lambda_{\max}$ also changes ignition speed: 3.3× at 0.1 versus 10× at 0.25–0.75. Within this sweep, the precise magnitude therefore has little apparent effect on peak Avg over $[0.1, 0.75]$, although

Table 9: **Multi-seed results** (avg@32). *3 seeds*: mean $\pm$ standard deviation over three training seeds. *Single run*: the value reported in Table 1 (Table 4 for τ_{down}=0.90).

Method	AIME25		HMMT25	
	3 seeds	Single run	3 seeds	Single run
Qwen3-4B-IT-2507				
+GRPO	57.3 ± 1.1	57.7	34.1 ± 1.2	34.1
+AntiSD	69.9 ± 1.2	70.2	46.5 ± 1.3	46.7
+AntiSD, τ_{down}=0.90	57.1 ± 1.6	56.5	40.3 ± 1.4	39.6
Olmo3-7B-TK				
+GRPO	71.5 ± 0.7	71.7	50.3 ± 0.9	50.5
+AntiSD	75.1 ± 0.8	75.3	55.6 ± 1.1	56.2

it changes the observed convergence speed and may affect peak Avg at the upper end. We claim robustness only for this interval and make no pairwise significance claims among its settings.

Table 10: **Sensitivity to** $\lambda_{\max}$ on Qwen3-4B-IT-2507 (single seed; five-benchmark Avg). GRPO reaches 51.3 Avg.

$\lambda_{\max}$	0.1	0.25	0.5 (canonical)	0.75	1.0
Avg	62.4	62.4	62.8	62.5	61.0
Speedup	3.3×	10.0×	10.0×	10.0×	5.0×

D.5 Privileged-context quality

To test how AntiSD depends on the quality of the privileged context, we replace the canonical outcome-verified peer rollout in c with degraded constructions on Qwen3-4B-IT-2507 (single seed; all else canonical; Table 11). *Dataset-only* is a constructed control that uses the bare dataset answer string; it is not a training fallback (Section 4). *Incorrect peer rollout* uses an incorrect rollout from the same group, and *shuffled* uses a solution to a different problem. Relative to the canonical 62.8 Avg, the three controls reach 61.9, 59.6 and 57.0. This single-seed pattern is consistent with context relevance modulating the benefit, but it is not causal confirmation. Even the shuffled context remains above GRPO (57.0 vs 51.3), so degraded c reduced the gain rather than causing observed harm in this run. Since about half of the canonical gain over GRPO (5.7 of 11.5 points) survives a solution to a different problem, part of the benefit does not require a problem-relevant solution trace; it may come from other content of c, such as the correctness feedback where retained, or from the reversed signal itself. This qualifies the No-teacher result in Section 4.3: removing the teacher collapses training, but not all of the gain depends on problem-relevant privileged content. The ablation probes end-to-end robustness to degraded c, not gate calibration in isolation; and the JSD transform caps the AntiSD advantage only on the negative-u side, so it is not a general safety bound for arbitrary corrupted contexts.

Table 11: **Privileged-context quality** on Qwen3-4B-IT-2507 (single seed; five-benchmark Avg).

Source of c	**Avg**
Outcome-verified peer rollout (canonical)	62.8
Dataset-only (bare answer string; constructed control)	61.9
Incorrect peer rollout	59.6
Shuffled (another problem's solution)	57.0
GRPO (reference)	51.3

D.6 Matched OPSD baseline: details

Our +SD baseline (Section 4) adds the sampled-token reverse-KL descent signal $\delta_t = +u_t$, computed with the current policy as teacher, to the GRPO advantage; it is a reward-anchored sampled-token baseline rather than a direct implementation of OPSD [38]. The matched OPSD baseline instead

implements OPSD's main objective on the same Qwen3-8B backbone, with the same DAPO-Math-17k prompt pool, 200-step horizon and five-benchmark evaluation protocol as Table 1: a frozen initial-policy teacher, pointwise-clipped forward KL, and full-vocabulary distillation, without a GRPO objective. DAPO-Math-17k provides final answers but no reference CoTs, so the privileged solution trace, when available, is an outcome-verified correct CoT sampled from the student's own rollout group; when no rollout is correct, the teacher receives correctness feedback without a solution trace. OPSD's original setting instead conditions the teacher on dataset reference solutions, and OPSD is reported to outperform GRPO there; this run tests OPSD's objective in our data setting and is not a reproduction of OPSD's reported results. Table 12 reports the AIME24, AIME25 and HMMT25 columns and the Avg over all five benchmarks. The matched baseline reaches 33.5 Avg, improving over +SD (30.6) but remaining below GRPO (57.4) and AntiSD (65.7). Because the teacher, divergence, vocabulary coverage and the presence of the GRPO term all change together, this comparison does not isolate the contribution of any single design choice. These results are consistent with independent reports of degradation or instability of standard self-distillation in challenging reasoning settings [9; 32; 13].

Table 12: **Matched OPSD baseline** on Qwen3-8B under the Table 1 protocol (single run). *Avg* is over all five benchmarks; the GRPO, +SD and AntiSD rows are those of Table 1.

Method	**AIME24**	**AIME25**	**HMMT25**	**Avg (all 5)**
+GRPO	73.5	65.2	39.2	57.4
+SD	40.1	30.5	14.9	30.6
+OPSD (matched)	43.6	33.7	15.6	33.5
+AntiSD	**78.4**	**73.4**	**54.4**	**65.7**

D.7 Sampled outcome coverage and modal-answer accuracy

To test whether improved pass@k coverage (Figure 3) comes at the expense of answer consistency, Table 13 reports maj@32 on HMMT25 at the best-Avg checkpoints of Olmo3-7B-IT, the model whose HMMT25 margin over GRPO is the smallest in Table 1. Both rows use the same 32 rollouts per problem; maj@32 is the accuracy of the modal normalised boxed answer among the parsed final answers, with ties scored by expected accuracy under a uniform random tie-break. AntiSD improves both pass@1 and maj@32 over GRPO, although the 3.3-point maj@32 difference is on the order of one HMMT25 problem and comes from a single model. Together with Figure 3, this indicates improved sampled outcome coverage without lower modal-answer accuracy; neither metric directly measures semantic novelty among reasoning paths.

Table 13: **Sampled outcome coverage and modal-answer accuracy** on HMMT25 for Olmo3-7B-IT at the best-Avg checkpoints.

Metric	Base	+GRPO	+SD	+AntiSD
pass@1 (avg@32)	25.9	31.2	24.4	**32.3**
maj@32 (↑)	33.3	40.0	30.0	**43.3**

D.8 Capability regression after math-only training

The Qwen3-8B checkpoints reported in Table 1 were trained only on math data. We therefore ask whether AntiSD *harms* general capabilities relative to GRPO under the same training setup. We evaluate the same Base, +GRPO, +SD and +AntiSD checkpoints on IFEval [40] (instruction following) and MMLU-Pro [28] (broad knowledge); Table 14. AntiSD's point estimate is not below GRPO on either benchmark. This is a check for capability regression after math-only training, not evidence of broad transfer or of equivalence: this small suite does not rule out smaller capability differences. MMLU-Pro is not a grounding benchmark, and we did not evaluate grounding, tool-mediated code execution or function calling.

Table 14: **Capability regression check** on Qwen3-8B checkpoints trained on math only.

Method	**IFEval**	**MMLU-Pro**
Qwen3-8B (base)	83.9	62.5
+GRPO	82.4	62.0
+SD	81.7	61.7
+AntiSD	83.6	62.7

E Limitations and Broader Impacts

Scope and extensions. The conditional-PMI account in Section 3.1 is a local, per-step characterization of the per-token signal rather than a global-optimum statement about the combined objective; understanding the long-horizon dynamics under the full ascent + gate update is itself an interesting question. Our evaluation spans five language models from the Qwen3 and Olmo-3 families (4B–30B parameters) on mathematical reasoning, with an initial single-model probe on code reasoning (Section 4.1, Table 3); our claims are scoped accordingly. Several natural extensions follow the same algorithmic skeleton (Algorithm 1) without modifying the AntiSD update: *(i)* broader code evaluation, which is the primary open item – more models and benchmarks, such as LiveCodeBench v6 with longer per-problem horizons and richer test-case structure; *(ii)* multi-turn agentic settings, where the reward depends on a sequence of tool calls rather than a single rollout, with the privileged context spanning the full interaction trace; and *(iii)* richer privileged-context content – process-level critiques, partial-credit annotations, and rationale-comparison rankings – replacing the binary or continuous correctness feedback used here. Larger model scales beyond 30B and multimodal conditioning are also natural settings to test whether the conditional-PMI characterization remains the dominant credit-assignment signal.

Mechanism evidence. Our mechanism evidence is token-level (Figure 2) and aggregate (Figures 3–5); we do not include same-prompt trajectory comparisons between trained +SD and AntiSD checkpoints. Because the two policies generate stochastic, non-token-aligned chains of thought, such side-by-side traces would serve as qualitative illustrations rather than controlled trajectory-level attribution.

Gate calibration. *(i) Threshold:* the multiplier $\tau_{\text{down}}/H_{\text{warm}} \in \{0.90, 0.93, 0.95\}$ has a substantial effect on Qwen3-4B-IT-2507 but little effect on peak Avg on Qwen3-8B, although 0.95 slows ignition there (Tables 4 and 7); 0.93 is the shared value used across the five evaluated models, not a per-model optimum or a claim of transfer beyond them. Changing it changes when the term deactivates; the separate H_{warm} condition governs reactivation. *(ii) When it matters:* whether the gate is needed at all is associated with initial teacher entropy in the runs tested – removing it materially affects the two Qwen runs (≈ 0.4 nats initially) but has little effect on the Olmo3-7B-IT run. How much its threshold value matters is a separate, model-conditional question that entropy does not explain, since both Qwen models start near 0.4 nats yet show very different sensitivity. Threshold values were swept directly only on these two Qwen models; using 0.93 on the other three models is not a cross-family threshold sweep. We have not separately varied benchmark difficulty, since the gate is calibrated on the shared training pool rather than per evaluation benchmark; recalibration was effective in the Qwen3-8B continual run (Section 4.4). *(iii) Under degraded c:* in the single-seed ablation of Appendix D.5, a solution to a different problem still yields 57.0 Avg against GRPO's 51.3, so the corrupted context reduced the gain rather than causing observed harm in that run; this probes end-to-end robustness to degraded c, not gate-calibration robustness in isolation.

Missing comparisons. We do not include a matched comparison against PRM-trained policies. A credible version requires training and validating a separate PRM and then rerunning policy training with matched backbone, training data, policy-training budget, group size and evaluation protocol, reporting the additional PRM-training compute separately; we regard it as an important next step.

Gate threshold. The gate state is dynamic, but the threshold itself is calibrated separately for each run as $\tau_{\text{down}} = 0.93\, H_{\text{warm}}$ and then held fixed after warm-up. A time-varying threshold could accommodate gradual shifts in the normal entropy level. However, a threshold updated too quickly could follow an entropy collapse downward and therefore fail to deactivate the AntiSD term, so a

dynamic version would require a slower or anchored reference and separate validation. We have not evaluated such variants and retain the simple fixed-after-warm-up rule.

Broader impacts. AntiSD is a post-training method that improves credit assignment in RLVR. Positive impacts: stronger open-weight reasoning models, lower training cost (2 to $10\times$ fewer steps to reach a given accuracy), and a clearer theoretical handle on why default self-distillation underperforms on math reasoning, which may inform future designs of process-level signals derived from a self-teacher. Negative impacts: as with any improvement to LLM reasoning, gains are dual-use; a stronger reasoning model can be applied to adversarial or harmful tasks. AntiSD does not introduce a new attack surface beyond the pre-existing dual-use profile of large language models, and we do not release new high-risk model artifacts. We see no specific path to fairness, privacy, or safety harms beyond those already attaching to the underlying open-weight base models.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Section 1 state three contributions: (i) the PMI characterization of the per-token signal in default self-distillation (Section 3.1), (ii) the AntiSD method based on JSD ascent with an entropy-triggered gate (Section 3.2), and (iii) experimental results across five models from 4B to 30B parameters on math reasoning, with an initial single-model code probe (Section 4.1), showing $2-10\times$ speedup and up to $+11.5$ pp final accuracy over GRPO (Section 4). Each claim maps directly to a section that supports it, and the scope of the claims (math reasoning; code as an initial probe) and their remaining limitations are stated in Section 6.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 6 contains a dedicated *Limitations* paragraph covering: the single-seed scope (three-seed results only for Qwen3-4B-IT-2507 and Olmo3-7B-TK on AIME25/HMMT25; all other results, including the matched OPSD, code-probe and continual runs, are single-seed); the hand-chosen 0.93 gate multiplier; the single-seed $\lambda_{\max}$ sweep, in which peak Avg varied little only over $[0.1, 0.75]$ while ignition speed varied; the model-conditional gate calibration; the dependence on the quality of the privileged context and the assumption of a verifiable reward and a privileged context (outcome-verified peer solutions plus correctness feedback); the math-reasoning scope with code as an initial probe; untested grounding, tool-mediated code execution, function calling and agentic settings, with IFEval/MMLU-Pro serving only as regression checks; and the missing matched PRM and same-prompt trajectory comparisons. Appendix E adds the local (per-step) nature of the PMI characterization, detailed gate-calibration guidance, the fixed-after-warm-up gate threshold, and the natural extensions that follow the same algorithmic skeleton.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution

is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.

- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Six lemmas and one remark underpin the method: the reverse-KL gradient identity (Lemma 1), the PMI characterization of u_t under self-distillation (Lemma 2), the trajectory-level potential-shaping telescope for the raw signal $\pm u_t$ (Lemma 3), the JSD f-divergence shape (Lemma 4), the properties of φ relied on in Section 3.2 (Lemma 5), the polarity of descent on a general convex f-divergence (Lemma 6), and the total-variation identity behind observation (O2) (Remark 7). Full assumptions and proofs are in Appendix A; each lemma is linked to the main-text equation or claim it supports.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Algorithm 1 (Appendix B) gives the per-step pseudocode; Table 6 (Appendix B) lists optimizer, learning rate, batch size, sequence lengths, sampling temperatures, gate calibration parameters, and hardware. Section 4 (Setup) and Appendix C specify the construction of the privileged context and provide the self-teacher prompt templates for math and code. Code is released at `https://github.com/FloyedShen/AntiSD`. Training data (DAPO-Math-17k, Dolci-RLZero) and base models (Qwen3, Olmo-3) are publicly released; the framework (verl) is open-source.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.

- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The implementation is released at `https://github.com/FloyedShen/AntiSD` under the Apache-2.0 license, including data preparation (`data/prepare_antisd.sh`), per-model and per-method training scripts (`recipe/antisd/run/`), and the AntiSD loss implementation (`verl/workers/actor/dp_actor.py`); training logs are available at `https://wandb.ai/brain-cog/AntiSD`. In addition, the paper provides Algorithm 1 (Appendix B), a complete hyperparameter table (Table 6), and self-teacher prompt templates (Appendix C); all base models and training/evaluation datasets are publicly released, and the underlying framework (verl) is open-source.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4 (Setup) specifies the training data (DAPO-Math-17k), number of steps (200), conditions (Base / +GRPO / +SD / +AntiSD), the privileged context (an outcome-verified correct peer rollout plus correctness feedback, or feedback only when no rollout in the group is correct; no dataset fallback), gate auto-calibration (H_{warm} is the median of the five warm-up step-level statistics and $\tau_{\text{down}} = 0.93\, H_{\text{warm}}$, a hand-chosen multiplier used for all models), and evaluation protocol (avg@32 on AIME 2024 / 2025 / 2026 and HMMT 2025; avg@4 on MinervaMath). Full hyperparameters, hardware, and sampling settings are in Appendix B (Table 6); self-teacher prompts are in Appendix C; the setups of the additional experiments (multi-seed runs, $\lambda_{\max}$ sweep, context-quality ablation, matched OPSD baseline, maj@32 and capability checks) are in Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Error bars cover only part of the evidence: the headline five-benchmark Avg gains and speedups (Table 1) are single-seed. We distinguish two sources of variability. Averaging over evaluation samples (avg@32 on AIME 2024 / 2025 / 2026 and HMMT 2025; avg@4 on MinervaMath) reduces evaluation-sampling noise but not training-seed variance. Training-seed variability is reported as mean $\pm$ one standard deviation (not the standard error) over 3 training seeds for Qwen3-4B-IT-2507 (GRPO, AntiSD, and AntiSD with τ_{down}=0.90) and Olmo3-7B-TK (GRPO, AntiSD) on AIME25 and HMMT25 (Table 2 and Appendix D.3). All other results (the remaining Table 1 settings and benchmark columns, the other component ablations, the $\lambda_{\max}$ sweep, the context-quality ablation, and the matched OPSD, code-probe and continual runs) are single-seed, as stated in Section 6; we do not run further training seeds because of compute (each 200-step on-policy RL run takes 10–75 hours on 16–32 H20 GPUs; Table 6). We make no formal significance tests.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix B (Table 6) specifies the hardware (8 NVIDIA H20 GPUs per node; 16 GPUs per run for 4B–8B models, 32 GPUs per run for Qwen3-30B-A3B), training duration (10–75 hours per 200-step run depending on model and sequence length), batch size, group size, maximum sequence lengths, and the verl framework [25]. Aggregate compute for the full project at submission time (reported experiments together with preliminary and discarded runs): approximately 1,000 H20-GPU-days, measured from wandb run-time logs. The additional 200-step runs added in this version (three-seed repeats, the $\lambda_{\max}$ sweep, the context-quality ablation and the matched OPSD baseline) are not included in this figure.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work is post-training of publicly released language models on publicly released math and code reasoning datasets, with no human subjects, scraped data, or new high-risk model artifacts. It conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Appendix E discusses both directions explicitly. Positive impacts: stronger open-weight reasoning models, lower training cost (2 to $10\times$ fewer steps to reach a given accuracy), and a clearer theoretical handle on default self-distillation failure on math reasoning. Negative impacts: gains are dual-use as with any improvement to LLM reasoning. AntiSD does not introduce a new attack surface beyond the pre-existing dual-use profile of large language models, and we do not release new high-risk model artifacts.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release new datasets or new pre-trained model artifacts; it describes a post-training method applied to existing publicly released base models on existing publicly released datasets.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All assets are cited and used within their terms; licenses are as stated on each asset's Hugging Face card or GitHub repository. Models: Qwen3-8B, Qwen3-4B-Instruct-2507, Qwen3-30B-A3B, Olmo-3-7B-Instruct and Olmo-3-7B-Think (all Apache-2.0). Code: verl (Apache-2.0). Training data: DAPO-Math-17k (Apache-2.0) and the Dolci-RLZero code data (ODC-BY). Evaluation: HumanEval+ and MBPP+ from EvalPlus (Apache-2.0), IFEval [40] (Apache-2.0) and MMLU-Pro [28] (MIT). AIME 2024/2025/2026 and HMMT 2025 are publicly available competition problems (the

MathArena HMMT release is CC BY-NC-SA 4.0), and the MinervaMath copy we use states no license; these sets are used for research evaluation only and are not redistributed.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The only new asset is the implementation code, released at `https://github.com/FloyedShen/AntiSD` with a README and an Apache-2.0 LICENSE; the paper documents it through full pseudocode (Algorithm 1, Appendix B) and hyperparameters (Table 6). The paper does not introduce new datasets or new pre-trained model artifacts.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The work does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The work does not involve human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: Large language models (Qwen3, Olmo-3) are the experimental *subject* of this work — the paper studies how to post-train them via on-policy reinforcement learning. The method's self-teacher is the policy LLM itself conditioned on privileged context; this use is described in Sections 2–3 and Appendix C. LLMs are not used as a separate component of the research methodology (e.g., not used as auto-evaluators, judges, data filters, or external sources of supervision). LLM use in writing or editing the paper itself, if any, falls under the NeurIPS exemption.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.