LEARNING WHAT IS WORTH TRAINING: THE DATA-VALUE BOTTLENECK IN LLM SELF-IMPROVEMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

As high-quality human data runs short, reasoning models are increasingly improved by training on their own verified solutions. These self-improvement loops usually stall after a few rounds, and the usual remedy is to generate more, or harder, verified data. We argue that the harder problem is deciding what to train on. The same verified data can help one model, do nothing for another, and damage abilities that should be kept, depending on the checkpoint it is applied to and on how it is used. We call this the *data-value bottleneck* and ask whether the value of a candidate update can be known before paying for a full training run.

In a pre-registered study of mathematical reasoning models from 1.5B to 27B parameters, we find that training value depends on both the current checkpoint and the training method. At a 9B checkpoint, a set of 322 problems had little effect when their solutions were used for supervised fine-tuning, but moved reinforcement learning one round ahead when the problems were used as prompts. The same teacher-generated solutions improved 4B and 9B checkpoints but reduced performance on protected tasks at 1.5B. None of the diagnostics we computed before training reliably distinguished useful updates from harmful or ineffective ones.

Short pilot runs were more informative about harm. Every candidate whose estimated value was negative after full training already had a negative estimated value after 25 optimizer steps, on the evaluation sets used to make the decision. The converse did not hold: some useful candidates also looked negative early, and pilots did not reliably rank candidates with similar final performance. We therefore propose a conservative veto-and-commit policy: reject candidates with evidence of harm, select a candidate only when its estimated value is credibly positive, and otherwise stop.

In four prospective tests, the policy never selected an update with negative value on a sealed evaluation, one that is read only once, after the decision is committed. This caution had a measurable cost: at a 2B checkpoint, the policy stopped even though the best candidate later produced a 5-point gain. It also prevented a much larger failure: at a near-ceiling 27B checkpoint, it rejected a teacher update that reduced sealed target accuracy by as much as 39 points after full training. We release the policy, evaluation protocol, code, pre-registration records and per-example results.

1 INTRODUCTION

Self-improvement systems for reasoning models often repeat a simple loop: sample solutions from the current model, keep those that can be verified, and train on them. Such loops stall within a few rounds (Song et al., 2024; Liu et al., 2026), and a common response is to look for harder problems or filter more carefully (Bae et al., 2026; Chen et al., 2025). This response assumes that the main limitation is the supply of useful training examples.

Our experiments point to a different practical problem. The same verified batch can help, do nothing, or cause regression depending on which checkpoint is updated and how the batch is used; the same

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

problems can be poor supervised targets and useful prompts for reinforcement learning (RL). The relevant question is therefore not only whether useful solutions can be generated, but whether a proposed training run will improve the target task without damaging capabilities that should be retained.

We call this the **data-value bottleneck**. A *state* is the checkpoint being updated, an *operator* is the training procedure together with its rollout and context settings, and *data* is the batch of examples or prompts supplied to that procedure. The value of an update is its gain on the target task after accounting for losses on protected tasks. In short, *support is not value*: being able to sample a correct solution does not imply that a particular training procedure can turn those samples into a reliable improvement. Existing work characterizes which behaviors a policy can produce (Mousavi-Hosseini & Erdogdu, 2026; Yue et al., 2025); we study which concrete update is worth running next.

We investigate this question in a pre-registered study of Qwen-family reasoning models from 1.5B to 27B parameters. Every prospective claim was frozen with a SHA-256 hash before the runs it predicts, and every miss is reported. Our contributions fall into three groups: what determines an update's value (C1, C2), how to decide whether to run it (C3), and how to evaluate it reliably (C4).

(C1) Training value depends on the checkpoint and the training method (§3). The same 322 verified problems are the weakest of six own-format supervised batches at a 9B checkpoint, yet as RL prompts at a matched 500-prompt dose they put the run a round ahead ($+7.0\,[+3.2, +10.8]$ at round two). The pool alone reaches the same level a round later, so the bank accelerates learning by one round without raising the final level. The same teacher rows help at 4B and 9B and cost preservation at 1.5B (three seeds), and the same kind of supply, rollouts sampled from each state's own base model, realizes $+2.9$ on a sealed suite at a 4B peak and $-10.8\,[-15.6, -6.0]$ at a 1.5B wall. Continuing from a supervised peak and retraining from the base on the same rows give opposite target–preservation trade-offs.

(C2) The diagnostics we tested before training do not predict update value (§4). Over 54 (case, surface) pairs at three scales, the geometry between the fresh-data gradient and the capability to be kept does not predict forgetting (Spearman $+0.31$, AUROC 0.62), and difficulty, outcome entropy and consensus alignment do not rank candidate batches. The 1.5B wall is not a supply shortage either: a quarter of its pool is still in the frontier band, yet RFT and GRPO extract nothing further.

(C3) In our tests, short pilots were useful for screening out harmful updates, but not for selecting the best candidate (§5). Across four prospective states and a 9B replay, all ten updates that later had negative value on the decision evaluations were already negative after 25 optimizer steps. However, four of the fifteen updates that ultimately had positive value also looked negative at that point. Pilot rankings were similarly unstable: for nine candidates at 9B, the correlation with full-run value fell from $\rho = 0.71$ on the original evaluation set to $\rho = 0.07$ on a fresh one. A negative pilot is therefore useful as a warning, but pilot scores do not reliably select the best candidate. We use this asymmetry to define a veto-and-commit policy. It rejects candidates with evidence of harm, commits only when a remaining candidate has a positive lower confidence bound, and otherwise stops. In four prospective tests it selected no update with negative value on the sealed evaluation. It missed a $+5.0$ update at 2B, avoided a pilot leader that later scored -1.4 at 1.5B, selected the empirical best action ($+2.9$) at 4B, and avoided a teacher update that collapsed at 27B.

(C4) We provide an auditable protocol for evaluating candidate updates (§6). Several implementation and evaluation choices produced score changes as large as the training effects being compared: cross-device offsets, prompt-format changes, adapters that appeared to load but were not applied, and evaluator batching (1.6 percentage points, pp, on one greedy checkpoint). We turn these failure modes into required checks: same-device paired comparisons, explicit adapter-effect tests, frozen prompt and inference settings, uncertainty reported across items, training seeds and evaluation configurations, and a sealed final evaluation. We release the implementation, evaluation contracts, pre-registration ledger and per-example records.

2 Problem Statement and Protocol

Objects. A *state* θ is a policy with (i) per-pool band statistics from K-sample rollouts—dead ($\hat{p}$=0), frontier ($0<\hat{p}<1$), mastered ($\hat{p}$=1), mean $\hat{p}$; (ii) surface scores with their sources; (iii) a

completion signature: mean length, the fraction of completions that revise a boxed answer, and first-boxed accuracy next to graded last-boxed accuracy. Coverage is not extractability: a prompt with success probability p under the policy yields a K-rollout group with any group-relative signal only with probability $I_K(p) = 1 - p^K - (1-p)^K$, so on-support prompts with small p are silent to a finite-group operator; the informative-group fraction we report is the empirical mean of this quantity. An *operator* m is a training procedure with a frozen recipe: fresh-LoRA supervised fine-tuning (SFT) (Hu et al., 2022) from a start checkpoint, iterative rejection fine-tuning (RFT), or GRPO (Shao et al., 2024) with the direct double-sided importance-sampling stabilizer of Hou et al. (2026) (GRPO+DIS): group-relative advantages, token ratios taken against the rollout engine's log-probabilities, tokens outside $(1-0.2,\ 1+0.28)$ masked from the gradient (asymmetric bounds as in Yu et al., 2025), a KL penalty β=0.04 to the round start, and two epochs per rollout batch (Appendix F). We treat the training contract—rollout, context and training windows and decode settings—as part of m, and condition every reported value on a frozen evaluation contract. These operators are representative, not exhaustive; preference optimization, PPO with a learned value model and full-parameter fine-tuning, among others, were not tested (Table 4), and every conclusion about operators is scoped to this set. A *candidate* D is a batch of rows or a prompt pool. The utility of an intervention is $V(D, m, \theta) = \Delta_{\text{target}} - \lambda \cdot \text{mean}_{s \in \text{OOD}} \max(0, -\Delta_s)$ over out-of-distribution (OOD) surfaces s with λ=1, target MATH-500 (Hendrycks et al., 2021; Lightman et al., 2024) and OOD surfaces AMC (83) and a same-family held-out set (100), all read greedily with a fixed boxed instruction. Retrospective tables use V; the prospective selection policy (§5) uses the worst-surface variant $U = \Delta_{\text{target}} - \lambda \max_s[-\Delta_s - \delta]_+$ with tolerance δ=2 pp, because a mean lets one surface's loss hide behind another's gain.

Models and data. A *peak* is the best checkpoint of an iterative RFT ladder; a *wall* is a state at which further rounds stop improving. Qwen2.5-1.5B (Qwen Team, 2024) (a wall state, the policy after six RFT rounds), Qwen3.5-2B (Qwen Team, 2026a) (an RFT peak, MATH-500 56.4), Qwen3-4B (Yang et al., 2025) and Qwen3.5-9B (Qwen Team, 2026a) (each at the peak of an iterative RFT ladder from its base); the post-trained Qwen3.8-27B (Qwen Team, 2026b) enters only as a prospective state, under its own chat template with thinking off. Inference uses vLLM (Kwon et al., 2023); coverage uses the unbiased pass@k estimator (Chen et al., 2021). Pool: 500 olympiad problems. Bank: 322 problems synthesized solution-first by a teacher model (a verified seed extended by three computational steps, the solution emitted as executable Python whose sandboxed run yields the gold answer) (Li et al., 2026); a contamination audit against every evaluation set found no shared 13-token window except one generic stem.

Discipline. Every arm and its control are read on one device under one engine contract; adapters are verified applied (base and adapter must differ on fixed prompts inside one engine); protocols are frozen before the runs they predict and deviations are timestamped addenda. Binomial standard error at n=500 is ≈ 2.1 pp; a same-device, same-contract greedy replicate is bit-identical, and changing only the batching setting moves it by 0.8–1.6 pp (§6).

3 Value Is Not in the Data

Demonstration 1: same rows, opposite value under two operators. From the 9B supervised peak (merged; readout 65.6 / AMC 33.7 / held-out 22.0), the 322-problem bank used as supervised continuation data is the weakest of six own-format batches ($V = +0.6$; Table 3). Added to the GRPO+DIS prompt pool from the same state (822 prompts against 500), it raises the three-round endpoint from 76.4 (pool only) to 81.8 ($+5.4$; AMC $57.8 \to 62.7$, held-out $44.0 \to 48.0$). The two arms differ in prompt count as well as content, so this is a same-rows contrast across operators, not a dose-matched one. The pre-registered dose-matched control (the bank replacing part of the pool at 500 prompts, seed 1042) and a same-seed pool-only arm separate content from volume (Figure 1a): composition is worth $+7.0\,[+3.2, +10.8]$ at round two and $+2.0\,[-1.8, +6.0]$ at round three (the pre-registered round-three clause is a miss), volume -0.2 and $+0.4$, and at round three the arms are within noise on AMC and held-out as well. Together, the matched controls show three things: rows worthless as supervised targets at this state are useful as RL prompts; their value does not come from prompt volume; and they speed training up by about one round rather than raising its final level, since the pool alone reaches the same accuracy one round later. The gain is extraction, not new coverage (Figure 1b): a 128-sample audit of the RL start, the 822-prompt endpoint and the base

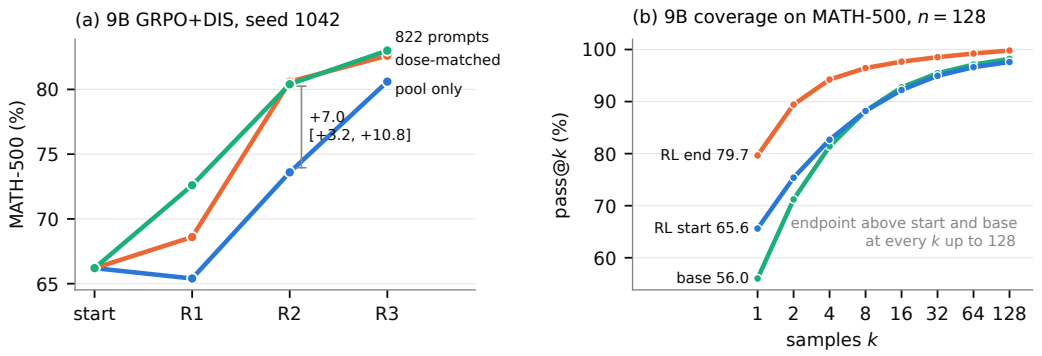


Figure 1: (a) The 322-problem bank as RL prompt supply at 9B: pool only (500 prompts), the bank replacing part of the pool at the same 500-prompt dose, and pool plus bank (822 prompts); one seed, one rig. The bank puts the run a round ahead and the pool alone catches up by round three. (b) Unbiased pass@k from 128 samples per item for the base, the RL start (supervised peak) and the RL endpoint.

Table 1: Demonstration 3: the identical 322 teacher-code rows as supervised continuation at three states (two epochs, one recipe; each arm read against its own merged start on one device). Paired per-item 95% intervals on the target; $V = \Delta MATH - 500-$ mean OOD drop. The 1.5B teacher row reports three-seed means; its paired target contrast is the seed-42 matched comparison against own rollouts.

state (start MATH-500/AMC/held-out)	supply	MATH-500 / AMC / held-out	ΔMATH-500 vs. own [95% CI]	V	seeds
9B peak (65.6 / 33.7 / 22.0)	own rollouts	69.0 / 32.5 / 30.0	—	+2.8	2
	teacher code	71.6 / 41.0 / 39.0	+2.6 [−0.8, +6.0]; s2: +4.0 [+0.4, +7.6]	**+6.0**	2
4B peak (62.6 / 42.2 / 38.0)	own rollouts	63.4 / 34.9 / 43.0	—	−2.9	1
	teacher code	66.0 / 42.2 / 38.0	+2.6 [−1.2, +6.4]	**+3.4**	1
1.5B wall (44.4 / 19.3 / 16.0)	own rollouts	45.8 / 18.1 / 16.0	—	+0.8	1
	teacher code	45.2 / 12.8 / 11.0	0.0 [−3.8, +3.6]	**−5.0**	3

shows pass@1 65.6 → 79.7 (base 56.0) but pass@128 97.6 → 99.8 (base 98.2), with no crossover up to k=128 and nested solvable sets.

Demonstration 2: continuation versus retraining. The iterative ladder's own recipe—a fresh LoRA retrained from the *base* on the growing cumulative set—compressed the 9B base's 3.1k-character reasoning to 1.6k and paid −8.5 (AMC) and −14 (held-out) at its first round, while lifting MATH-500 62.2 → 67.0 over two rounds and then stopping (66.2). *Continuing* from the peak with a fresh LoRA on 600 of the policy's own correct rollouts improved MATH-500 in every one of six batches (+0.6 to +4.4) with held-out at or above the start, AMC within 1.2 points of it or above, and no length change. Continuing on the full cumulative set recovers held-out (+6.0) but still pays −4.8 on AMC: the breadth tax is data×recipe, and its largest share belongs to the retrain-from-base step. Two batches replicated with a second seed land within ±2.5 with order preserved.

Demonstration 3: same rows, three states (Table 1). The teacher's program-of-thought solutions beat the policy's own rollouts at 9B on two seeds and at 4B on one, without a breadth tax, and at the 9B peak recover most of the base's held-out accuracy (22 → 39). At the 1.5B wall the identical rows leave the target exactly unchanged (47 problems each way) and lose 6.5 points of AMC and 5.0 of held-out on average, with losses on every seed: target-neutral and preservation-negative. What these support is the *sign* of the state effect: at n=500 the 9B and 4B point estimates are directionally positive with one of three intervals excluding zero, and the 1.5B profile differs in kind, not in degree.

The RL operator is state-dependent too. At 9B, three GRPO+DIS rounds from the peak break the supervised ceiling—67.0 → 72.8 → 81.8 with the bank, 64.8 → 65.8 → 76.4 without—through a late re-expansion into long verify-and-revise reasoning: revising completions 2% → 84%, length ×3, first-answer accuracy *falling* 67 → 61 while final accuracy jumps, in every 822-prompt series at round three regardless of launch point, and at or above the base on all five surfaces (AMC 39.8 → 62.7). The re-expansion is not what carries the level: the dose-matched arm (500 prompts,

Table 2: Three interventions that increased the gradient signal of one GRPO+DIS round without increasing target value. Each row compares two runs from the same state that differ in one factor; ΔT is the paired difference on the 1000-item decision target with a 95% item-bootstrap interval.

intervention (vs. control)	state	informative groups	ΔT (pp) [95%]	sealed
dense partial-progress credit (vs. outcome)	1.5B wall	$60 \to 254$ of 320	$+0.2\,[-1.8, +2.3]$	$+1.9$ vs. $+1.1$
hard OpenR1 frontier (vs. original pool)	27B, 3072	$0.33 \to 0.46$	$-0.8\,[-1.7,\ \ 0.0]$	$+0.5\,[-0.5, +1.5]$
6144-token window (vs. 3072), hard pool	27B	$0.46 \to 0.69$	$+0.6\,[-0.1, +1.4]$	–
6144-token window (vs. 3072), original pool	27B	$0.33 \to 0.30$	$-0.5\,[-1.2, +0.2]$	–

the bank replacing part of the pool; seed 1042, H100) reaches $66.2 \to 68.6 \to 80.6 \to 82.6$ with revising completions at 0.2%, first-answer accuracy equal to final (82.4 vs 82.6) and length $\times 1.4$, while the 822-prompt arm under the same seed ($66.2 \to 72.6 \to 80.4$ by round two) is already revising 17% of completions with first-answer accuracy three points below final. Accuracy gains and changes in reasoning behavior can occur independently: the behavior change tracks the larger pool and step count, the accuracy gain tracks the bank. At 4B, from its own peak, the same operator raises accuracy without any re-expansion (revising completions $\leq 0.8\%$, length within 10%) over two independent seeds, and its first-round gain is $+6.2$ for one seed and $+1.2$ for the other (5.0 pp apart, paired CI excluding zero), while the two-round totals agree ($+6.4$ and $+5.2$): a one-round readout measures the seed as much as the action. Sampled support explains the split's direction: at the 4B start only 2.2% of rollout mass is a beneficial revision (first answer wrong, final right) and GRPO shrinks it ($0.0217 \to 0.0125$) while ordinary correctness grows (dead problems $53\% \to 39\%$)—amplification of already-sampled trajectories, consistent with but not proof of the mechanism. At 1.5B four RL rounds show no transition. The same operator on comparable supply has three different values at three states, and at one of them its value is a distribution five points wide.

Demonstration 4: more gradient signal is not more value. Headroom is surface-specific: the post-trained 27B looks saturated on MATH-500 (86% of problems solved in all 16 samples, $\mathbb{E}[I_8] = 0.09$) but not on the 500-problem training pool (34% frontier, 0.24), whereas the 9B pool never saturated across three RL rounds ($\mathbb{E}[I_8]$ between 0.59 and 0.71; Figure 2a). A 1358-problem OpenR1 hard pool refreshes the supply further (44% frontier, 0.37). Signal still did not become value. Widening the rollout window from 3072 to 6144 tokens, with all other engine settings matched, cut truncation on the hard pool from 62% to 31% and raised its informative-group fraction from 0.46 to 0.69; on the original pool it *lowered* the fraction ($0.33 \to 0.30$), because a short window turns some otherwise correct long completions into failures and so creates mixed-reward groups (Figure 2c). Decision-surface utility stayed between -1.3 and -0.5 in all four cells, and the pre-registered interaction was $+1.1\,[0.0, +2.3]$ (Figure 2d). On 281 held-out hard-frontier problems three of the four cells gained 3–4 pp (intervals through zero): harder supply moved the policy toward the hard distribution without value on the saturated target (Appendix H).

Across all three interventions in Table 2, more signal did not become more value. Dense partial-progress credit at the 1.5B wall quadrupled the gradient-carrying groups, and a refreshed hard pool and a longer window raised the 27B's informative fraction by 0.13 and 0.24; no intervention moved the target by more than one point, and every interval includes or touches zero. At these states the binding constraint is not the amount of gradient signal but whether the update carries value for the target (per-cell details in Table 6).

4 Static Predictors Fail in Our Setting

Frontier availability does not diagnose the wall. Across three archived 1.5B ladders (RFT, GRPO, GRPO+DIS; twelve state→gain transitions), training-pool frontier mass is flat or rising while round gains decay from $+3.8$ to -0.2; measured at a common $K{=}16$, its pooled correlation with the next gain is -0.49—the wrong sign for the supply thesis—and no supply-side quantity predicts gain (Appendix A). The wall round is an extraction stall: the final RFT round moves pool $\hat{p}$ by -0.001 with a quarter of the pool still in the frontier band.

Gradient geometry does not predict forgetting (pre-registered). For 27 cases (a start policy and the fresh data actually trained on it: the 9B and 4B ladders, the 9B GRPO series, twelve 1.5B arms,

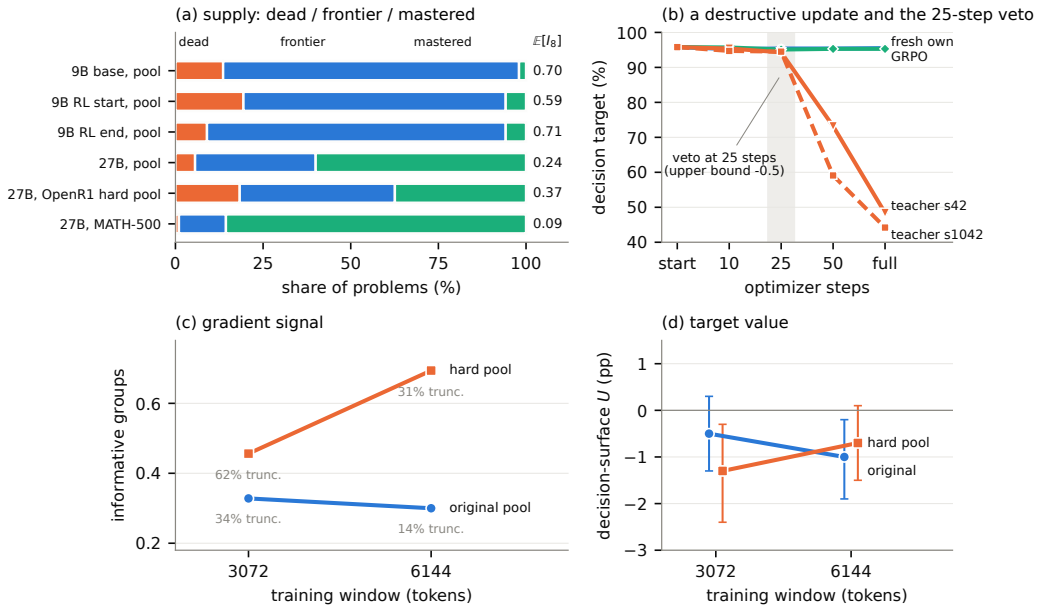


Figure 2: The post-trained 27B (chat template, thinking off). (a) Supply bands at 16 samples per problem (8 for the hard pool) and the expected informative-group fraction $\mathbb{E}[I_8]$, against the 9B RL series. (b) Decision-target accuracy along the pilot schedule for the prospective actions: teacher-code SFT (two seeds) collapses after the 25-step veto point, while own-rollout SFT and GRPO stay within a point of the start (the two lines overlap). (c, d) The pre-registered 2×2 of supply and training window for one GRPO+DIS round: informative-group fraction with generation-cap truncation, and full-run decision-surface utility with 95% paired intervals.

and three dose arms sharing fresh data) with 54 measured held-out outcomes, we computed at the start policy the cosine, negative-projection ratio, and layer-wise sign agreement between $\nabla_\theta L(D_f)$ and $\nabla_\theta L(A_s)$, where the anchor A_s is the policy's own correct completions on surface s; the baseline is centroid cosine distance in representation space. Fresh and anchor gradients are positively aligned in every pair (cosine 0.06–0.81; negative projection $\approx$ 0 in 50 of 54), including the pair with the largest forgetting (held-out -19). Pooled Spearman with the realized drop is $+0.31$, AUROC 0.62 (thresholds 0.5/0.75: miss); within-scale signs disagree; the 9B fresh gradient is *more* aligned with AMC than with GSM8K (Cobbe et al., 2021) yet AMC fell and GSM8K rose. Post hoc, the realized LoRA update ΔW of every trained adapter is orthogonal to the anchor gradient ($|\cos| \leq 0.04$, 40/40) and to the fresh gradient itself, and its first-order Taylor term predicts an anchor-loss *decrease* in every supervised case, the rounds that lost 13–19 points as much as those that gained 6–7 (a post hoc analysis; the pre-registered predictor test is Figure 5 in Appendix B). A low-rank update lives outside the span of the gradients computable before training; forgetting here is a change of reasoning mode (§3), not a conflict of directions. Semantic distance separates forgetting within 1.5B (AUROC 0.85) and inverts at 9B (0.19); the dose control behaves as designed (identical geometry, drops of -4.2, -3.6, -9.0).

Cheap signals do not rank candidates. Over the six own-format batches of Table 3, mean $\hat{p}$, informative-group fraction, outcome entropy, target length, and a DIEM-style consensus alignment (Tan et al., 2026a) each have $|\rho| = 0.34$ with full-run utility, consensus negative; the DIEM ablation (alignment $\gg$ difficulty) does not reproduce as a batch-selection signal. Our prior beliefs about "bad" data were wrong twice: the policy's own failed traces do no harm ($+1.4$) and the teacher's code was the best batch.

Table 3: Nine candidate batches from the 9B peak (start $65.6/33.7/22.0$): V after 10/25/50 steps and after two epochs; full scores on MATH-500/AMC/held-out. Pilot-vs-full Spearman: $+0.50$ (10), $+\mathbf{0.71}$ (25), $+0.34$ (50); the worst batch is in the pilot bottom two at every length.

batch	V_{10}	V_{25}	V_{50}	V_{full}	full scores
random	$+1.0$	$+1.8$	$+2.2$	$+2.8$	69.0 / 32.5 / 30.0
easy ($\hat{p} \geq 0.75$)	-0.4	$+0.6$	$+1.0$	$+4.4$	70.0 / 37.3 / 27.0
boundary	$+0.6$	-0.0	$+3.4$	$+2.6$	68.2 / 37.3 / 28.0
bank as own rollouts	-0.0	-0.6	$+0.4$	$+0.6$	66.2 / 37.3 / 29.0
consensus-top	$+2.2$	$+1.0$	$+2.6$	$+2.2$	67.8 / 41.0 / 24.0
consensus-bottom	$+0.6$	$+1.0$	$+4.0$	$+2.6$	68.2 / 38.6 / 26.0
failed rollouts as targets	-1.0	$+0.8$	$+3.2$	$+1.4$	68.2 / 31.3 / 26.0
stale (base-policy) rollouts	-0.6	-0.6	-2.0	$-\mathbf{3.6}$	63.2 / 31.3 / 30.0
teacher code solutions	$+1.4$	$+2.2$	$+3.0$	$+\mathbf{6.0}$	71.6 / 41.0 / 39.0

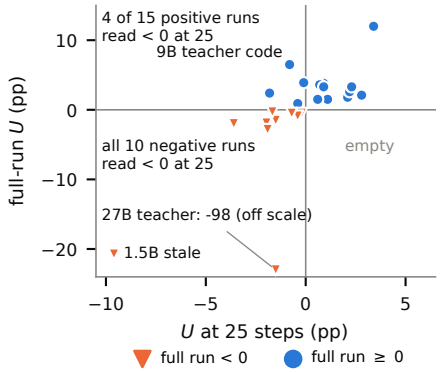
5 Measured Interventions: What Short Pilots Can and Cannot Do

Design (pre-registered in three addenda). From the merged 9B peak, nine 600-row candidate batches: random, easy ($\hat{p} \geq 0.75$), boundary, the bank as own-rollout targets, consensus-top and -bottom, and three we predicted harmful—the policy's own failed rollouts, the base policy's two-round-stale rollouts, and the teacher's code solutions. Each trained as a fresh LoRA for 10, 25, 50 optimizer steps and two epochs (≈ 150 steps), same recipe, each checkpoint read on three surfaces.

Pilots detect harm more reliably than they rank. On the nine 9B batches the pilots rank well on the surface the rule was built on ($\rho = +0.50$, $+\mathbf{0.71}$ and $+0.34$ at 10, 25 and 50 steps; Table 3), and the stale batch (-3.6; -5.2 on a second seed) is in the pilot bottom two at every length. Re-reading the same checkpoints on decision surfaces never used to build the rule removes the ranking (Spearman $\rho = 0.19$, 0.07 and 0.63 between conservative pilot and full-run utility): the realized best batch, teacher code ($+6.5\,[+3.5, +8.9]$), reads -0.8 at 25 steps, and at 4B the same teacher rows read -4.8 at 25 steps before finishing at $+3.4$, a transient format cost. The sign of a harmful update is more stable than any ranking (Figure 3, left): across the four prospective states and the 9B replay, all 10 updates that ended negative were negative at 25 steps, and so were 4 of the 15 that ended positive. Sample size explains why ranking fails: at 25 steps the paired half-width of $\Delta MATH - 500$ is 2.2 pp on 500 items and 6.7 pp on the 83–100-item protected sets, against pilot effects of 1–4 pp, and a paired-bootstrap selection rule abstained at every horizon on the nine batches (Appendix G). Reading each snapshot with four sampled completions at T=0.6 halves the half-width (1.3–2.0 pp against 2.3–3.9) and ranks five 2B candidates against their realized utilities with $\rho = 0.90$.

A veto-and-commit policy (Figure 4). Snapshots at 10, 25 and 50 steps are read on paired decision surfaces of ≥ 300 items disjoint from the scoring surfaces. At 25 steps the policy *rejects* a candidate whose paired upper bound is below zero or whose point utility is below -0.5 pp at both 10 and 25, *commits* the survivor with the highest positive lower bound, and otherwise repeats at 50 steps, then stops. We compare four policies against the empirical oracle, the best action on a sealed suite read once after the decision is committed: V (this rule), V80 (V with one-sided 80% bounds), S (commit only when the leader's lower bound exceeds every rival's upper bound, otherwise abstain, operationally STOP) and P (the 25-step point argmax). V, S and P were declared before the first prospective test; V80 was introduced after the 2B result and is prospective only for the later states. Thresholds, decision surfaces (1000/499/300 items, audited disjoint) and the sealed procedure were frozen before each test. Some mechanism arms (dense credit, the 27B refresh arm, a second 2B GRPO seed, a reweighting control) were also read on the sealed suite after the lock; they enter no decision.

Prospective outcomes. No policy committed an update with negative sealed utility (Figure 3, right). The certificate rules were too conservative at the 2B peak, where three candidates had sealed gains of $+4.2$ to $+5.0$ with intervals excluding zero and only the point rule captured one, and right at the 1.5B wall, where the point rule realized -1.4 and rollouts from the 1.5B base model were



state	empirical oracle	V	S	P
2B peak	variants +5.0	STOP (5.0)	abstain (5.0)	variants (0)
1.5B wall	GRPO +1.4	STOP (1.4)	abstain (1.4)	fresh own (2.8)
4B peak	base rollouts +2.9	base roll. (0)	abstain (2.9)	base roll. (0)
27B post-trained	fresh own +0.3	STOP (0.3)	abstain (0.3)	fresh own (0)
9B replay	teacher code +9.3	easy (4.7)	abstain (9.3)	bank (6.6)

Sealed utility U in pp; regret against the empirical oracle in parentheses. V80 matches V except at 2B, where it commits variants (it was declared after that state). No policy committed a destructive action; 27B teacher code realized -78.8.

Figure 3: Left: utility after 25 steps against full-run utility for all 25 (state, action) cells of the four prospective states and the 9B replay (decision surfaces, conservative over seeds); the 27B teacher cell (-98) is drawn at the axis. Right: prospective decisions and the sealed utility of the best observed action, with each policy's regret (full table: Table 5).

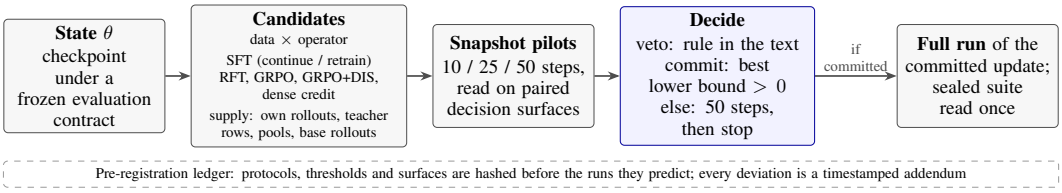


Figure 4: Schematic of the decision loop evaluated in this paper; the exact veto rule is given in the text. A candidate pairs a data batch or prompt pool with one of the operators we studied (Table 4); methods outside that set were not tested. Pilots are prefixes of the full run, so they cost only readouts ($\sim$4 GPU-minutes per snapshot at 9B).

$-10.8\,[-15.6, -6.0]$. At the 4B peak the same kind of supply, rollouts from the 4B base model, was the oracle ($+2.9$) and V committed it, although it cost 7.2 points on the MMLU-Pro transfer surface on both seeds; the utility charges the 5.2 beyond tolerance, and an 8.1–8.5-point target gain outweighs it. At the post-trained 27B, whose decision surfaces sit at 95.8–97.0%, the oracle was indistinguishable from STOP, and teacher-code supervision was vetoed at 25 steps, when its decision-target accuracy had moved by only -1.5 and -1.3 points on the two seeds but its paired upper bound was already negative; it then fell to 73 and 59 at 50 steps and to 48 and 44 at the end, -78.8 on the sealed suite (Figure 2b, Appendix H).

Does state-conditioned selection beat a fixed action? Over the 3×3 matrix of states $\{1.5\text{B}, 4\text{B}, 9\text{B}\}$ and actions $\{\text{STOP}, \text{own-rollout SFT}, \text{teacher-code SFT}\}$ at matched budgets, the per-state oracle averages $+3.60$ and the best fixed action (teacher code) $+1.86$, a difference of $+1.74$ pp with 95% interval $[+0.00, +3.61]$, carried entirely by the 1.5B row and below the 5.0 pp seed range of one action at 4B, so we do not claim that choosing per state beats a fixed intervention in this action set.

6 Evaluation Safeguards Are Part of the Method

Four hazards each inverted a conclusion during this work, and each is now a gate in the released harness. **(1) An adapter reported as loaded was not applied:** vLLM 0.21 re-keys PEFT adapters for Qwen3.5 hybrid checkpoints with a mapper that drops them silently, so our first 2B and 9B ladders were base-model readouts (500/500 identical completions). Gate: base and adapter must differ on fixed prompts in one engine. **(2) Device offsets:** identical adapters differed by up to 2.4 pp across two rigs and by 1.2 pp across two GPUs on one machine. Gate: one device per table. **(3) Format drift:** the same 4B adapters rise $58 \rightarrow 65$ without a boxed instruction and fall $61 \rightarrow 55$ with it. Gate: one prompt protocol per ladder. **(4) Batching:** one 9B adapter reads 67.8 or 66.2

greedily at `max_num_seqs` 16 or 24, with 18 of 500 answers flipping. Gate: the batching setting is frozen with the prompt protocol.

Three-layer uncertainty. Every headline delta carries (i) a paired per-item bootstrap interval, (ii) the range across training seeds where more than one exists, and (iii) the range across evaluator contracts where more than one was run. Layer (i) alone—the usual report—understates the uncertainty of this regime: the 4B first-round GRPO gain is $+6.2$ $[+3.0, +9.6]$ on one seed and $+1.2$ $[-2.0, +4.4]$ on another. Merging a LoRA into bf16 weights costs 1.4 pp at 9B and 3.8 pp at 1.5B, so arms are compared against the merged start's own readout.

7 Related Work

Why self-improvement stalls. Stalls have been attributed to verifier error (Song et al., 2024) and to finite-sample decay at a fixed distribution (Liu et al., 2026); our ladders use gold rewards and still stall. Difficulty filtering and frontier synthesis (Bae et al., 2026; Chen et al., 2025) raise frontier mass, which never predicted gain here, and rise-and-collapse under self-training (Lin, 2026) matches our post-peak regime. He et al. (2026) assign skills to SFT or RL by mastery; we add that how the supervised update is applied (continuation or retraining) can flip its sign. **Data selection within a fixed recipe.** DIEM (Tan et al., 2026a) reweights prompts by consensus inside an RL step, and its signal did not transfer to selecting batches before training. DataFlex-RL (Liang et al., 2026) finds that no selection or reweighting policy beats uniform sampling under one GRPO recipe and that dropping one evaluation domain reverses the method ranking, which is why our utility takes the worst protected surface and our sealed suite includes a transfer domain. Both study a data policy at a fixed operator and state; we vary both and include STOP. **Coverage and extractability.** Whether RLVR surfaces or creates capability is contested (Yue et al., 2025); Mousavi-Hosseini & Erdogdu (2026) prove that outcome-reward policy gradient can only boost prompts with non-trivial base likelihood. Our K-sample bands are a coarse outcome-level proxy for that coverage, and our 1.5B wall sits where coverage is non-zero yet RFT and GRPO extract no gain. Dense partial-progress credit (Fu et al., 2026) is the theoretically licensed route past silent groups; our re-implementation at the wall quadrupled the signal without value (Table 2). The exploration-potential score of Yue et al. (2026) is the published form of our informative-group statistic. **Evaluation in self-improving systems.** Xu et al. (2026) show that single-decode ledgers inflate gains, and MetaRSI (Tan et al., 2026b) keeps the evaluator outside every write surface, as our sealed suite does. Frontis-MA1 (Yang et al., 2026) composes trained operators with a search harness, and Dream-RSI (Zheng et al., 2026) makes exploration decisions replayable over recorded outcomes; an unexecuted weight update has no recorded outcome, and snapshot pilots approximate that replay. Solution-first synthesis follows Li et al. (2026).

8 Limitations and Conclusion

All states are Qwen-family math-reasoning models trained with LoRA and the operators of Table 4; the post-trained 27B uses a separate chat and training contract and is not part of the matched cross-scale comparisons. Most ladder rows and the 9B RL series are single-seed (the 9B RL series has a second seed only on another rig), and the policy has four prospective tests and one replay with one or two seeds per action, too few for a general methods claim. The action set is small, and the static negatives are bounded by our predictors and 32-row gradient batches.

We asked whether the value of a self-improvement update can be known before it is run, and our answer has three parts. First, value is not a property of data: the same verified rows changed sign with the checkpoint and the training method, and none of the diagnostics we computed before training could tell the cases apart. Second, short pilots screen for harm but do not rank: every update that ended negative on the decision surfaces was already negative at 25 steps, while close candidates could not be separated at practical sample sizes. Third, a policy built on that asymmetry (veto, commit only on credible gain, otherwise stop) never committed an update with negative sealed utility across four prospective states, at the price of sometimes leaving a modest gain unused. For practitioners, each candidate update is an intervention whose value should be measured on the current model, under a frozen evaluation contract, before it is run.

Reproducibility statement

The supplementary material contains the `dvb` package (evaluation contracts, decision policies, trainers and rollout generator; the trainer objective is in App. F), campaign recipes, every pre-registration protocol with the freeze ledger, the decision and sealed surfaces, and per-item records from which `dvb reproduce` recomputes every prospective decision and sealed score on a CPU (points and decisions exactly, interval endpoints within 0.5 pp). State checkpoints are omitted for size.

Ethics statement

The study uses publicly released math benchmarks and datasets under their published licenses and involves no human subjects or personal data. Model-generated training data (the teacher-written bank and solutions) was checked by executing its solutions or comparing them with reference answers before use. The work aims to make self-improving training safer by detecting harmful updates before they are committed; it studies open models and introduces no new capability beyond them.

Use of generative AI

We used generative AI tools to propose and refine hypotheses, draft experimental plans and pre-registration protocols, critique drafts, implement and run the training, evaluation and analysis code, prepare datasets, analyze and interpret results, find and summarize related literature and verify references, translate, and write and edit text. Model-generated content is also part of the method (the teacher-written bank and solutions, and the dense-credit judge). The paper contains no formal theorems or proofs. All numbers come from executed code and are traceable to the released per-item records; the authors directed the study, reviewed the protocols, results and interpretations, and take full responsibility for the paper.

References

Sanghwan Bae, Jiwoo Hong, Min Young Lee, Hanbyul Kim, Jeongyeon Nam, and Donghyun Kwak. Online difficulty filtering for reasoning oriented reinforcement learning. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 700–719, 2026. doi: 10.18653/v1/2026.eacl-long.30.

Mark Chen et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Xuanzhong Chen, Zile Qiao, Guoxin Chen, Liangcai Su, Zhen Zhang, et al. AgentFrontier: Expanding the capability frontier of LLM agents with ZPD-guided data synthesis. *arXiv preprint arXiv:2510.24695*, 2025.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Preston Fu, Kevin Frans, Oleh Rybkin, Sergey Levine, and Aviral Kumar. Long-horizon language model reinforcement learning via progressive point matching. *arXiv preprint arXiv:2609.07303*, 2026.

Chongyang He, Rui Zhang, Zixuan Wang, and Xin Li. Learning what to learn: Stage-specific data sets for SFT-then-RL in small language model reasoning. *arXiv preprint arXiv:2606.04466*, 2026.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2021.

Zhenyu Hou, Yujiang Li, Jie Tang, and Yuxiao Dong. Single-rollout asynchronous optimization for agentic reinforcement learning. *arXiv preprint arXiv:2607.07508*, 2026.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, 2023.

Zhongzhi Li, Yucheng Shi, Zongxia Li, Ruhan Wang, Anhao Li, Zixun Huang, Junyao Yang, Lei Ke, Ninghao Liu, Haitao Mi, and Leowei Liang. Recursive synthesis for long-horizon terminal tasks. *arXiv preprint arXiv:2608.05466*, 2026.

Hao Liang, Mingrui Chen, Hengyi Feng, Meiyi Qiang, and Wentao Zhang. DataFlex-RL: An evaluation platform for RLVR data policies. *arXiv preprint arXiv:2609.06107*, 2026.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, 2024.

Jianzhe Lin. Self-improvement can self-regress: The rise-and-collapse failure mode of LLM self-training. *arXiv preprint arXiv:2606.21090*, 2026.

Chenruo Liu, Yijun Dong, Yiqiu Shen, and Qi Lei. A task-centric theory for iterative self-improvement with easy-to-hard curricula. *arXiv preprint arXiv:2602.10014*, 2026.

Alireza Mousavi-Hosseini and Murat A. Erdogdu. Post-training with policy gradients: Optimality and the base model barrier. In *International Conference on Machine Learning (ICML)*, 2026. arXiv:2603.06957.

Open R1. OpenR1-Math-220k. `https://huggingface.co/datasets/open-r1/OpenR1-Math-220k`, 2025.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Qwen Team. Qwen3.5: Towards native multimodal agents. `https://qwen.ai/blog?id=qwen3.5`, 2026a.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork. `https://qwen.ai/blog?id=qwen3.8`; model card `https://huggingface.co/Qwen/Qwen3.8-27B`, 2026b.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Yuda Song, Hanlin Zhang, Carson Eisenach, Sham M. Kakade, Dean Foster, and Udaya Ghai. Mind the gap: Examining the self-improvement capabilities of large language models. *arXiv preprint arXiv:2412.02674*, 2024.

Haoru Tan, Sitong Wu, Yanfeng Chen, Shizhen Zhao, Yang-Tian Sun, Tianjia Liu, Chirui Chang, Shaofeng Zhang, Samm Sun, Xiuzhe Wu, Ruobing Xie, and Xiaojuan Qi. Dynamic important example mining for reinforcement finetuning. *arXiv preprint arXiv:2608.29252 (CVPR 2026)*, 2026a.

Zihan Tan, Leixin Sun, Zitong Shi, Yitao Liu, Jiajun Wu, Nathaniel Brooks, Jiaru Qian, Xiaoran Shang, Suyuan Huang, Yi Ding, et al. MetaRSI / RSI2: A meta-recursive self-improving system for recursive self-improving systems themselves. *arXiv preprint arXiv:2609.06396*, 2026b.

Yubo Wang et al. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2024.

Cheng Xu, Nan Yan, Liming Chen, and M-Tahar Kechadi. Phantom gains: Auditing self-improvement against a measured null. *arXiv preprint arXiv:2608.20290*, 2026.

An Yang et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Junlin Yang, Che Jiang, Yu Fu, Tianwei Luo, Can Ren, Weizhi Wang, Kaikai Zhao, Hongyi Liu, Yuxin Zuo, Yuru Wang, Yuchen Fan, Kai Tian, Zhenzhao Yuan, Xiaojian Lin, Li Sheng, Rushi Qiang, Guoli Jia, Xingtai Lv, Ermo Hua, Dianqiao Lei, Youbang Sun, Ning Ding, Bowen Zhou, and Kaiyan Zhang. Frontis-MA1: Training an AI4AI model towards recursive self-improvement in machine learning engineering. *arXiv preprint arXiv:2607.28568*, 2026.

Qiying Yu et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1933.

Yuanhao Yue, Qianli Ma, Chengyu Wang, Haoting Wang, Lei Shen, and Jun Huang. Not all prompts are equal: Exploration-guided prompt scaffolding for multimodal reinforcement post-training. *arXiv preprint arXiv:2609.15051*, 2026.

Tong Zheng, Xidong Wu, Zheng Zhang, Zhankui He, Chaoyi Zhang, Benjamin Coleman, Ruoqiao Wei, Di Bai, Haolin Liu, Rui Liu, Xue Wang, Yue Zhuan, Wang-Cheng Kang, Renkai Xiang, Heng Huang, Xinwu Cheng, and Yunsong Guo. Dream-rsi: Recursive self-improvement through evolving worlds. *arXiv preprint arXiv:2609.14858*, 2026.

A THE WALL: RETROSPECTIVE AND TWO GAIN CHANNELS

Three archived Qwen2.5-1.5B ladders from one warm-start (MATH-500 43.8): iterative RFT ($K{=}4$) reaches 51.2, GRPO 46.4, GRPO+DIS 51.0 at round four (seed 42); three-seed endpoints 50.6 ± 0.5, 46.4 ± 1.4, 49.7 ± 1.7 (AMC 18.9 ± 5.7, 13.7 ± 1.9, 19.7 ± 1.8). Probing all 13 checkpoints at $K{=}16$: under RFT the eval-side frontier falls ($0.464 \to 0.354$) as mastery rises (absorption); under GRPO greedy MATH-500 climbs $+2.6$ while no sampled statistic moves (sharpening). Predictors of next-round gain at $K{=}16$ (Pearson, $n{=}12$): train frontier mass -0.49, train absorption -0.27 (sign-reversing across ladders), held-out mastered -0.76, eval-side frontier $+0.55$; descriptive only. At 9B the sampled distribution itself moves under RL (mean $\hat{p}$ $0.54 \to 0.80$, pass@16 $0.65 \to 0.93$ on a 100-problem subset), so "GRPO = sharpening" is a property of the 1.5B wall state.

B GRADIENT GEOMETRY DOES NOT PREDICT FORGETTING

The pre-registered static-predictor test of §4: the realized held-out drop of every (case, surface) pair against the gradient-alignment predictor and a semantic-distance baseline.

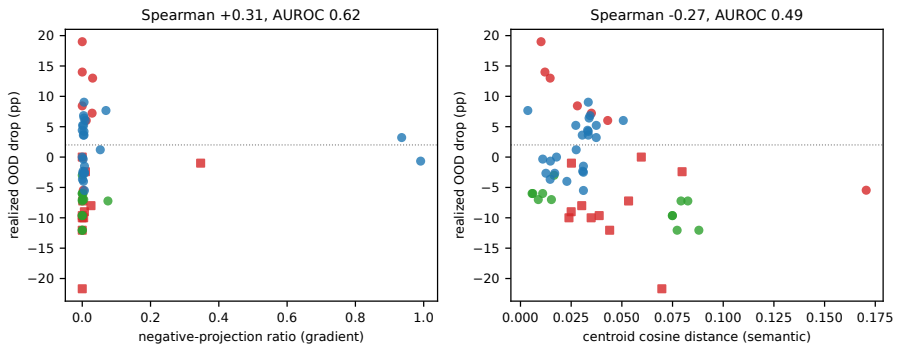


Figure 5: Realized held-out drop against the gradient predictor (left) and the semantic baseline (right), 54 (case, surface) pairs at 1.5B (red), 4B (green), 9B (blue); squares are GRPO cases.

C DUELS, DOSE, REGIMENS, AND THE CONTROLLER AT 1.5B

Pre-registered equal-budget duel at the wall state (three seeds; same-rig control 49.6 / 27.7 / 14.0 on MATH-500/AMC/held-out): synthesized novel problems $+1.7$, exogenous novel $+1.3$, stale re-mining $+0.7$ (σ 2.2), self-distillation on stale rollouts -3.1 (3/3 destructive, de-mastery signature). Mixture sweep monotone in the easy-novel fraction ($+0.73 \rightarrow +3.40$); volume sweep with an interior optimum near 14% of the mix ($+2.4 \rightarrow +3.4 \rightarrow +1.6 \rightarrow +0.8$), the three surfaces identifying dose as a distribution-migration dial (novelty-family surface $+5.5$, AMC -9.0 at the top dose); family-agnostic at moderate dose (GSM8K-family rows $+3.3$). Regimens (accumulate, replace, consolidate-then-dose) all stay in the 53–54 band. A six-rule measured-state controller reached equal-or-better endpoints than fixed iteration at half the rounds (two seeds; v2 with in-loop synthesis, three seeds) and its STOP verdicts were vindicated by the baseline's subsequent rounds; the durable lesson is argmax-checkpoint delivery.

D SCALE LADDERS AND THE 9B RL SERIES

Corrected ladders (uniform boxed protocol, one device each): Qwen3.5-2B 38.0, 50.0, 55.2, 56.4, 55.2 (off-band novelty at R4 55.0/54.4); Qwen3-4B 45.8, 61.0, 58.2, 56.6, 54.8, 58.6, 58.0 (novelty at R6 58.0/58.8; GRPO+DIS from R6 58.2); Qwen3.5-9B 62.2, 65.8, 67.0, 66.2. 9B GRPO+DIS from the peak: pool+bank 67.0, 72.8, 81.8 (first-boxed 61.2; revising 84%; 5.5k chars); pool only 64.8, 65.8, 76.4; from R1 then +bank 69.0, 69.4, 78.6. Breadth (base / SFT peak / RL3 pool / RL3 pool+bank; RL rig, where the peak reads held-out 24.0 against 22.0 on the supervised rig): AMC 39.8 / 33.7 / 57.8 / 62.7; held-out 43.0 / 24.0 / 44.0 / 48.0; AIME24 16.7 / 3.3 / 30.0 / 30.0; GSM8K 82.7 / 88.2 / 88.4 / 92.3. 4B GRPO from the peak (H100 rig; R0 63.6): seed 42 69.8, 70.0, 71.2; seed 2042 64.8, 68.8, 69.0: the seeds differ at round one ($+5.0$, CI $[+1.4, +8.6]$) and are indistinguishable by round two ($+1.2$, CI $[-2.2, +4.6]$); revising $\leq 0.8\%$ throughout. 1.5B GRPO+DIS from the wall: 53.8, 53.6, 55.4, 54.4; no transition. At 9B, off-policy GRPO (rollouts from the base) stays at 65.8: GRPO pays only on the policy's own samples. **Seed-1042 replication (pre-registered).** On one rig, the seed-1042 arms run about a round ahead of the archived seed-42 arms (another rig) and reach similar round-three levels: 822 prompts $72.6 \rightarrow 80.4 \rightarrow 83.0$ against $67.0 \rightarrow 72.8 \rightarrow 81.8$, pool only $65.4 \rightarrow 73.6 \rightarrow 80.6$ against $64.8 \rightarrow 65.8 \rightarrow 76.4$. Applied retrospectively to the 4B Track-B batches, the policy stops rather than selecting the wrong batch (both 25-step intervals include zero). **Large-k coverage audit (post hoc development read; not pre-registered).** 128 samples per item at T=0.6 on MATH-500 for the RL start state (the merged supervised peak), the round-three 822-prompt endpoint and the base, unbiased pass@k (start / endpoint / base): k=1 65.6 / 79.7 / 56.0; k=4 82.7 / 94.2 / 81.4; k=16 92.2 / 97.7 / 92.7; k=64 96.6 / 99.2 / 97.1; k=128 97.6 / 99.8 / 98.2. There is no $k \leq 128$ at which the start or the base overtakes the endpoint. Solvable sets at n=128: start-only 0.0%, endpoint-only 2.2%; base-only 0.0%, endpoint-only 1.6%; unsolved by all three 0.2% (one item). Items never solved in 128 draws: start 12, endpoint 1, base 9; items solved in every draw: 70 / 61 / 0—the base's support is broad and never certain. Expected 16-draw bands (dead / frontier / mastered): base 7.3 / 87.1 / 5.7% ($\mathbb{E}[I_8]$=0.74), start 7.8 / 60.2 / 32.1% (0.49), endpoint 2.3 / 62.0 / 35.7% (0.49). Success-probability quantiles (10th / 20th / 50th): start 0.07 / 0.21 / 0.80, endpoint 0.43 / 0.62 / 0.91, base 0.09 / 0.20 / 0.66: the transition lifted the low tail. Of the 61 start items with $\hat{p} < 0.1$, 18 reach $\hat{p} \geq 0.5$ at the endpoint and 16 stay below 0.1; the 228 start-frontier items move from mean $\hat{p}$ 0.52 to 0.76. Mean completion length 1.7k / 4.9k / 3.0k characters. Under the large-k standard the 9B transition is extraction inside an essentially fixed support: pass@1 moved $+14.1$, pass@128 $+2.2$.

E PRESERVATION OPERATORS: ANCHORED CONTINUATION ACROSS SCALES

A cached top-K logit forward-KL anchor on the warm-start's own correct completions (dual-ascent β to a KL target δ=0.005; anchor batch N_a=30, non-monotone in size: N_a=10 and 100 both worse) is the preservation operator behind the "anchored" arms. On Qwen3-8B at an SFT warm-start (64.6 / 47.0 MATH-500/AMC), continued SFT on fresh data gains $+6.0$ on the target and loses 7.2 on AMC in a matched pair; the anchored run keeps $+5.4$ with AMC $+1.2$; over five seeds the control is 70.7 ± 0.7 / 45.1 ± 3.5 and the anchored run 70.6 ± 1.0 / 46.0 ± 2.7 —the single-seed $+8.4$

on AMC shrinks to $+0.9$, the same seed lesson as §6. Anchored means are at or above control on AMC and held-out at 8B, 1.7B and 1.5B (3–5 seeds) and neutral or negative on off-family GSM8K (1.7B: $47.4 \rightarrow 38.9$): anchors protect the distribution they contain. Weight decay (44.6 AMC) and a lower learning rate (45.8, losing the target gain) do not substitute. An anchor drawn from the fresh-data distribution is destructive (β escalates 6×10^4-fold). Anchors preserve and do not improve uncovered benchmarks (AIME $10.0 \rightarrow 10.0$ with AIME-correct anchor rows), and more informative fresh data inverts the anchor's marginal value (with Nemotron traces the control already preserves AMC at 47.0; anchor -3.6). Group-signal probes across scales: informative groups 10.8–16.8% in 1.5B GRPO caches, 2–15% at 9B/27B on hard and GSM8K pools; a 50-problem probe estimates the full statistic within $\sim$4 pp.

F Training Operators Studied and the GRPO+DIS Objective

Table 4 lists every training operator used in the paper. All train LoRA adapters (rank 16 on the attention projections) from the stated start; none is a full-parameter update.

Table 4: Training operators studied. Conclusions about operators are scoped to this set.

operator	supply	where
SFT, continued from the current checkpoint (fresh LoRA)	own correct rollouts, teacher-code rows, variants, stale or base rollouts	all states; §3, §5
SFT, retrained from the base on the cumulative set	iterative-ladder rows	9B ladder; §3
iterative rejection fine-tuning (RFT)	the policy's own verified rollouts, repeated	1.5B, 2B, 4B, 9B ladders; App. D
GRPO (group-relative, no importance correction)	prompt pool	1.5B duels; App. C
GRPO+DIS (objective below)	olympiad pool, teacher bank, OpenR1 hard pool	1.5B, 2B, 4B, 9B, 27B; §3, §5
GRPO+DIS with dense partial-progress credit	prompt pool, judged per segment	1.5B wall; Table 2
off-policy GRPO (rollouts sampled from the base)	prompt pool	9B; App. D
anchored continuation (top-K logit KL anchor)	own correct rollouts	8B, 1.7B, 1.5B; App. E

Training methods not studied: preference optimization (DPO and variants), PPO with a learned value or reward model, full-parameter fine-tuning, distillation from teacher logits, model merging.
Settings outside scope: non-math domains and multi-turn or agentic tasks.

The GRPO+DIS objective. For a prompt x the rollout engine samples K completions $y^{(1..K)}$ at temperature T and records the per-token log-probabilities $\log \pi_{\text{roll}}(y_t^{(k)})$ it sampled them with; rewards $R_k \in \{0, 1\}$ come from the boxed-answer grader. Advantages are group-relative, $\hat{A}_k = (R_k - \bar{R})/(\sigma_R + 10^{-4})$, and rows with $|\hat{A}_k| \leq 10^{-3}$ (all-pass and all-fail groups) are dropped. With $r_{k,t} = \exp\big(\log \pi_\theta(y_t^{(k)}) - \log \pi_{\text{roll}}(y_t^{(k)})\big)$ and a stop-gradient weight $w_{k,t} = \text{sg}\big[r_{k,t}\, \mathbf{1}\{1 - \varepsilon_\ell < r_{k,t} < 1 + \varepsilon_h\}\big]$,

$$\mathcal{L}(\theta) = -\frac{1}{N} \sum_{k,t} w_{k,t}\, \hat{A}_k \log \pi_\theta\big(y_t^{(k)}\big) + \beta\, \frac{1}{N} \sum_{k,t} \Big(\log \pi_\theta\big(y_t^{(k)}\big) - \log \pi_{\text{ref}}\big(y_t^{(k)}\big) \Big),$$

with N the number of completion tokens in the batch. Tokens whose ratio leaves the asymmetric trust region are masked from the gradient rather than clipped (the direct double-sided importance-sampling stabilizer of Hou et al. (2026); the asymmetric bounds follow Yu et al. (2025)). Constants: $\varepsilon_\ell = 0.2$, $\varepsilon_h = 0.28$, $\beta = 0.04$, π_{ref} the round-start policy, learning rate 10^{-5} with 20 warm-up steps, LoRA rank 16, per-device batch 1 with 8-step accumulation, two epochs over each round's rollout batch; the training window is part of the contract (3072 tokens at the 27B unless stated). We borrow only the DIS stabilizer: rollouts are synchronous K-sample groups with group-relative advantages, not SAO's single-rollout asynchronous estimator. One implementation detail departs from this definition: for the base-model states the trainer scored completions under the supervised prompt format, which omits the boxed instruction the rollouts were sampled with, so r_t compared slightly different contexts; the trust region then masked 3–4% of completion tokens (mean ratio 0.99–1.00), against 0.02% at the post-trained 27B, whose trainer reused the rollout's prompt tokens. The released trainer always reuses the rollout prompt tokens. The PPM arm (App. G) replaces $\hat{A}_k$ by a per-token advantage and is otherwise identical.

G Pilot Power and the Prospective Test

Retrospective (development, not validation). The lower-confidence-bound rule (λ=1, δ=1, ε=0.5, ε_{big}=3, 1000 paired draws; this development read used δ=1, the prospective tests δ=2)

on the nine 9B batches: at 10 steps the leader (random) has $U = +1.0\,[-1.0, +2.6]$ against a best other upper bound of $+4.0$; at 25 steps teacher code $+2.2\,[-1.4, +4.2]$ against $+4.0$; at 50 steps consensus-bottom $+4.0\,[-5.0, +7.0]$ against $+7.4$: abstain, abstain, abstain, STOP. Realized full-run U: teacher code $+6.0$, easy $+4.4$, random $+3.2$, …, stale -3.8 (never rejected: upper bounds $+2.0/+1.8/+0.6$). Regret against the oracle 6.0 pp; the rule equals always-STOP. **Power.** Paired discordance (items whose correctness flips between start and checkpoint) at 25 steps is 6.6% on MATH-500 (SE 1.1 pp, half-width 2.2), 8–11% on AMC (SE 3.4, half-width 6.7) and 11–13% on held-out (half-width 6.7); after two epochs discordance reaches 14–25% and half-widths 3.3–11 pp. A 200-item decision set gives ± 3.5 pp on the target alone. **Prospective test (P2B, frozen 2026-09-18 before the first step).** State: the merged 2B RFT peak (MATH-500 56.4 on its rig). Actions: STOP; stale continuation (the state's own training rows); fresh own-rollout SFT; teacher-code SFT; off-band variant SFT (each 320 rows, 80 steps, snapshots at 10/25/50, seeds 42 and 1042); GRPO+DIS for one round on 320 frozen prompts. Decision surfaces: 1000 MATH-train items, 499 in-domain protected items, 300 GSM8K items, audited disjoint (exact and word-8-gram Jaccard) from every scoring surface and every action pool. Scoring: a sealed 1000/500/500 suite (MATH target, MATH near, MMLU-Pro (Wang et al., 2024) STEM transfer) read once after the decision file is committed. Three pre-declared policies are scored against always-STOP, every fixed action, random and the oracle: veto-and-commit (primary), the strict lower-bound rule above, and point-argmax at 25 steps. **Outcome (2B).** Decision reads (two seeds): stale and one-round GRPO had negative upper bounds at 25 steps and were rejected; variants led with $U_{25} = +2.8\,[-0.4, +5.2]$, fresh own-rollouts $+2.1\,[-0.8, +4.4]$, teacher -0.4. V stopped, S abstained; P chose variants. Sealed suite (read once after the committed lock): θ 51.0 / 52.6 / 13.2 on target / near / transfer; realized U: variants $+\mathbf{5.0}\,[+2.4, +7.5]$, teacher $+4.3\,[+1.6, +7.0]$, fresh own $+4.2\,[+1.7, +6.8]$, GRPO $+1.4\,[-0.3, +3.2]$, stale $-0.5\,[-2.3, +1.3]$. Regret: V and S (abstaining) 5.0 (captured 0%), P 0 (100%), random action (uniform over non-STOP actions) $+2.9$ expected. No policy committed a destructive action. Post hoc, the one-sided 80% variant commits variants at 25 steps (lower bound $+0.8$). A protocol deviation is recorded: the GRPO arm first failed with an out-of-memory error, an automation error committed a four-action lock and started a sealed pass that was killed before any output; the amended five-action lock preceded every sealed score. **Outcome (1.5B wall).** θ 54.3 / 56.9 / 70.7 on the decision surfaces (sealed 52.1 / 56.4 / 14.8). Base-rollout supply was rejected at 25 steps (upper bound -4.4; full-run decision read -20.7), teacher at -0.3; GRPO and fresh own-rollouts survived without a positive bound. V and the 80% variant stopped, S abstained; P chose fresh own-rollouts ($U_{25} = +0.7$). Sealed: GRPO $+1.4\,[-0.4, +3.0]$, teacher $+0.6\,[-2.3, +3.0]$, fresh own $-1.4\,[-5.4, +1.3]$, base rollouts $-10.8\,[-15.6, -6.0]$; regret 1.4 for the stopping rules, 2.8 for P; no destructive commit. **4B peak.** Decision reads (two SFT seeds, one GRPO seed): V, V80 and P select the base-rollout supply (V lower bound $+1.20$, 80% bound $+2.00$, $U_{25} = +3.6$; fresh own $+2.5$, teacher $+2.4$, GRPO $+0.6$); S abstained. Sealed (θ 62.4 / 65.8 / 42.6): stale $+2.9\,[-1.1, +7.1]$, teacher $+2.1\,[-0.4, +4.2]$, GRPO $+0.6\,[-1.3, +2.2]$, fresh own $+0.3\,[-1.8, +2.5]$; V, V80 and P captured the oracle, S regret 2.9; no destructive commit. **Readout power (2B, development).** Separately trained 25-step pilots and step-25 snapshots agree within their intervals (fresh own U $+1.0$ vs $+2.1$; stale -0.2 vs -0.4). A second GRPO seed (rollout seed 1001, training seed 1042) reads 0.0 on the decision surfaces and $+2.0$ sealed against -1.9 / $+1.4$ for seed 42; a DIEM-reweighted GRPO control kept 55 of 320 groups and reads -0.7 / $+0.8$. Sampled reads (K=4, T=0.6, 499 protected items): half-widths 1.3–2.0 pp against 2.3–3.9 greedy; $\rho(K4_{25}, \text{realized}) = 0.90$. A deviation is recorded: the job's on-node decision step crashed on a stale module, the release gate opened on an empty hash, and a sealed read of θ started and was killed within six minutes with no output; the lock was recomputed locally and committed before the sealed job. **9B peak (Level 1: actions pre-existed; surfaces unseen).** Decision reads of the nine Track-B batches (θ 64.3 / 67.3 / 86.3): V abstains at 25 steps (best lower bound -0.71) and commits easy at 50 (lower bound $+2.0$; full-run decision read $+3.9\,[+1.8, +5.9]$); the 80% variant agrees; S abstains; P picks the bank batch ($U_{25} = +1.1$, full $+1.5$). The decision-read oracle is teacher code ($+6.5\,[+3.5, +8.9]$), which read -0.8 at 25 steps; stale supply is $-2.8\,[-6.5, -0.1]$ and sat in the pilot bottom two at every horizon without crossing the veto's upper bound ($+0.2$ at 25). Frozen predictions: S abstains everywhere (hit); V commits one of {teacher code, easy, random} or abstains (hit, via easy at 50); P commits teacher code (miss). Sealed (θ 59.7 / 64.6 / 16.2): teacher code $+\mathbf{9.3}\,[+6.7, +11.9]$, consensus-bottom $+5.4$, boundary $+5.2$, consensus-top $+5.0$, easy $+4.6\,[+2.4, +6.8]$, random $+3.9$, failed targets $+3.2$, bank $+2.7\,[+0.4, +4.9]$, stale $-0.7\,[-3.9, +1.3]$: V and the 80% variant captured 49% of the oracle (regret 4.7), P 29%, S none; no destructive commit; the oracle is the batch the 25-step pilot read most negative.

Table 5: Prospective states (sealed suite, read once after each committed lock). Utility U with 95% paired intervals; regret against the empirical oracle; S abstentions are operationally STOP. The 9B row is a Level-1 replay (actions pre-existed; surfaces unseen). No policy committed a destructive action at any state.

state	empirical oracle (U)	V	V80	S	P
2B peak	variants $+5.0\,[+2.4, +7.5]$	STOP (5.0)	variants† (0)	abstain (5.0)	variants (0)
1.5B wall	GRPO $+1.4\,[-0.4, +3.0]$	STOP (1.4)	STOP (1.4)	abstain (1.4)	fresh own (2.8)
4B peak	base rollouts $+2.9\,[-1.1, +7.1]$	base roll. (0)	base roll. (0)	abstain (2.9)	base roll. (0)
27B post-trained	fresh own $+0.3\,[-0.9, +1.4]$	STOP (0.3)	STOP (0.3)	abstain (0.3)	fresh own (0)
9B (replay)	teacher code $+9.3\,[+6.7, +11.9]$	easy (4.7)	easy (4.7)	abstain (9.3)	bank (6.6)

†V80 was declared after the 2B read: post hoc at 2B, prospective at the later states.

Dense credit at the wall (mechanism duel, pre-registered). Same 1.5B state, the same 2560 rollouts and recipe as the P15B GRPO arm; only the credit changed: progressive-point-matching segment advantages against 3–6 reference reasoning points per prompt extracted from the gold solutions (292 of 320 prompts), judged per rollout by a fixed model (calibration: mean fraction of points reached 0.78 for correct rollouts vs 0.31 for incorrect). Informative groups rose from 60 to 254 of 320 and gradient-carrying rows from 480 to 2029 (P1 hit); the target moved $+0.2\,[-1.8, +2.3]$ relative to outcome credit retrained on the same node (P2 miss), preservation stayed within tolerance (P3 hit), length did not grow; sealed: $+1.9\,[-0.1, +3.6]$ vs $+1.1\,[-2.0, +2.7]$. The wall is not primarily a credit-assignment bottleneck.

H A Post-Trained 27B State: Headroom, Supply, and the Prospective Test

Every ladder above starts from a base checkpoint. Qwen3.8-27B (a hybrid-attention 27B, released only post-trained) is read under a separate contract—its own chat template, thinking off unless stated, boxed answer instruction, 8k-token context, one H200—and is reported here as its own table; nothing in it is comparable to the base-derived ladders. All predictions were frozen before the first read (§I). **Headroom (Tier A).** MATH-500 greedy 96.2 (2.9k characters; thinking on 98.0); MATH-500 at K=16, T=0.8: mean $\hat{p}$ 0.959, dead 1.0%, frontier 13.4%, mastered 85.6%, $\mathbb{E}[I_8] = 0.085$—saturated. On the 500-problem olympiad pool that carried the 9B RL series: mean $\hat{p}$ 0.839, dead 5.6% / frontier 34.4% / mastered 60.0%, $\mathbb{E}[I_8] = 0.235$, $\mathbb{E}[I_{16}] = 0.291$, mean EPS 0.040; thinking on: 0.858, 5.4 / 25.8 / 68.8%, $\mathbb{E}[I_8] = 0.171$. Predictions: A1 (greedy ≥ 85) hit; A2 (mastered $\geq 60\%$, dead $\leq 10\%$) hit; **A3 ($\mathbb{E}[I_8] \leq 0.15$, "saturated, STOP") miss**: a third of the pool is still frontier for the 27B, so outcome-reward GRPO on the existing pool is not silent by construction. A4, read on the pre-registered pool at K=16 under one contract, hit and separates the scales: expected frontier band 34.4% at the 27B against 85.2% at the 9B RL endpoint, 84.4% at the 9B base and 74.8% at the 9B supervised peak, with $\mathbb{E}[I_8]$ 0.235 against 0.71 / 0.70 / 0.59: three GRPO rounds took the 9B's mean pass rate on this pool from 0.35 at the RL start to 0.54 (base 0.43) without consuming the supply, and the supervised peak reads *below the base* there (0.349 against 0.431, dead items 19.4% against 13.6%): the ladder's target gain came at the cost of coverage on the pool it was about to train on. The 27B is the only state in that table whose supply is mostly mastered. The dead set is the same size in both modes (5.6 vs 5.4%): thinking moves items from the frontier into the mastered band and does not open the dead band. **Supply refresh (Tier A2).** A 1358-problem hard pool (OpenR1-Math (Open R1, 2025), competition sources, verified integer or short answers, audited disjoint from every decision and scoring surface) read at K=8: mean $\hat{p}$ 0.631, dead 18.3% / frontier 44.3% (601 problems) / mastered 37.4%, $\mathbb{E}[I_8] = 0.370$, 8.4k characters; S1–S3 (mean $\hat{p} < 0.8$, frontier ≥ 400, $\mathbb{E}[I_8] \geq 0.30$) all hit. The two reads are the point of §3 at a new state: the same rows that had value for the 9B RL series are 60% mastered for the 27B, and their value has to be measured again at the new state. **Prospective test (Tier B) and refresh arm.** Under the frozen gate (pool $\mathbb{E}[I_8] \geq 0.10$) Tier B runs on the original pool with the chat-formatted actions of §G (own-rollout SFT, teacher-code SFT, one GRPO+DIS round on 320 prompts, STOP; V/V80/S/P; sealed suite read once after the committed lock); a pre-declared refresh arm trains the same GRPO recipe on 320 of the 601 hard-pool frontier problems, outside the frozen decision set, to compare GRPO(pool) with GRPO(refreshed) at the same state (predictions R1–R3: $\geq 1.5\times$ informative groups, $\geq +1.0$

pp on the target, preservation within δ). **Outcome.** Decision reads (θ 95.8 / 96.6 / 97.0; conservative over seeds): own-rollout SFT $U_{25} = -0.4\,[-1.4, +0.5]$, GRPO $-0.7\,[-1.6, +0.2]$, teacher $-1.5\,[-3.1, -0.5]$ and then -65.8 at 50 steps and -98.1 full (dev1000 $95.8 \to 48.4$ and 44.2 on the two seeds); V, V80 STOP; S abstain; P own-rollout SFT. Sealed (θ 93.8 / 94.2 / 60.0): own-rollout SFT $+0.3\,[-0.9, +1.4]$ (oracle), GRPO $0.0\,[-1.0, +1.0]$, teacher $-78.8\,[-84.0, -73.5]$; regret 0.3 for the stopping rules, 0 for P, random (uniform over non-STOP actions) -26.2; no destructive commit. Teacher's -78.8 is the conservative (worse-seed) value: seed 1042 lost 39.4 on the target and 41.4 on the near surface (39.4 beyond δ), but only 1.4 on the MMLU-Pro transfer surface; seed 42 lost 22.2 / 23.2 / 1.2 ($U = -43.4$). Prior (STOP or teacher; own-rollout and GRPO null): the nulls and STOP hit, teacher missed. **Training contract.** The 3072-token window (fixed for one-GPU memory) truncated 34% of pool rollouts and 62% of refreshed ones; the correct rate among untruncated rollouts was 0.92 and 0.63, the latter equal to the 8k-window read. The window changes signal in opposite directions by supply: training-time informative groups exceed the read-time $\mathbb{E}[I_8]$ (pool 0.33 vs 0.235; refreshed 0.46 vs 0.37) because truncation fails long correct rollouts, and only 12 of 105 and 43 of 146 informative groups are truncation-free. A pre-registered 2×2 (supply $\times$ window $3072/6144$, all other engine settings matched) lowers truncation to 14% and 31% (L1 hit), moves the informative fraction to 0.30 on the pool and 0.69 on the refreshed supply (L2 hit), and moves the refreshed arm's target read by only $+0.6$ ($-1.3 \to -0.7$; L3 miss); difference-in-differences $+1.1\,[0.0, +2.3]$. R1 (informative $\geq 1.5\times$: $1.39\times$) and R2 (refreshed $-$ pool on the target: $-0.8\,[-1.7, 0.0]$ decision, $+0.5\,[-0.5, +1.5]$ sealed) miss; R3 hits. On 281 held-out frontier problems (θ 48.0) three of four cells gain 3–4 pp (intervals through zero at this n) while the pool/3072 cell loses 2.1: harder supply moved the policy toward the hard distribution without value on the saturated target. The longer window did not restore detectable target value at the pre-registered threshold (the interaction interval reaches zero): harder supply added support and gradient signal but no detectable target value.

Table 6: The 27B training-contract 2×2 (GRPO+DIS, one round, seed 42; all other engine settings matched). Truncation is at the generation cap; informative = groups with non-zero advantage; U on the decision surfaces (full run) with 95% paired intervals; held-out = 281 hard-frontier problems not used in training (θ 48.0).

	pool / 3072	pool / 6144	hard / 3072	hard / 6144
truncated rollouts	33.9%	13.8%	62.4%	31.1%
informative groups	0.328	0.300	0.456	0.694
of which truncation-free	12	24	43	77
completion tokens trained	5.04M	6.89M	6.18M	9.98M
decision U (full)	$-0.5\,[-1.3, +0.3]$	$-1.0\,[-1.9, -0.2]$	$-1.3\,[-2.4, -0.3]$	$-0.7\,[-1.5, +0.1]$
sealed U	$0.0\,[-1.0, +1.0]$	–	$+0.5\,[-0.5, +1.5]$	–
held-out Δ	-2.1	$+3.2$	$+3.6$	$+3.9$

I Pre-registration Ledger

Every prospective claim was frozen (SHA-256 of the protocol text + UTC timestamp) before the training it predicts; amendments are timestamped addenda before affected runs. 26 protocol documents scoring over 70 clauses; roughly 40% reversed or falsified, each miss carrying its diagnosis. Track A: A1 miss, A2 vacuous, A3 miss, A4 miss, A5 miss on sign (orthogonal). Track B: B1 miss, B1′ hit on nine batches (miss on six), B2 miss, B3 miss, B4 hit; B-cont C1 miss (AMC clause), C2 hit; B-bad (i) miss, (ii) hit at 25 steps / miss at 50, (iii) miss, (iv) miss; B-seed2 both hit; B-4B T4-1 hit, T4-2 miss, T4-3 miss; B-1.5B T1-1 miss, T1-2 miss. Scale/RL: S9B-1 hit, S9B-2 hit, gentle marginal, 2B rescue miss; RL9B-R2-1 miss, -2 hit, -3 hit/miss, -4 miss; F1 miss (replicate), F2 miss, F3 hit; RL1.5B-3 hit; RL9B-BR hit; RL9B-LAT miss. Prospective campaigns (frozen 2026-09-18 onward; 46 freeze entries in total): P2B prior (stale destructive, GRPO null-to-negative, variants and teacher positive, fresh own null): stale -0.5 not destructive (miss), GRPO $+1.4$ (miss on sign), variants/teacher hit, fresh own $+4.2$ (miss); P15B prior (V does not commit teacher; fresh own or GRPO the only committable candidate, or V stops) hit; P4B prior (V commits teacher or abstains; P differs from V) miss on both clauses; P9B Level 1: V clause hit, S hit, P miss; PPM P1 hit, P2 miss, P3 hit; P27B A1 hit, A2 hit, A3 miss, A4 hit (same pool), thinking-on clauses hit, S1–S3 hit, Tier B prior STOP/nulls hit and teacher miss, R1 miss, R2 miss, R3 hit, L1 hit, L2 hit, L3 miss, M1 hit on

the point estimate (interval through zero); RL9B dose D1 miss, D2 miss, D3 unscored. Hashes and per-item records ship in the supplementary material.