HARP: REUSING TEACHER RL HISTORIES TO CONSOLIDATE REASONING AND LONG-HORIZON AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning produces experts that solve problems and experts that carry out long-horizon tasks, while deployment calls for one policy that does both. Such consolidation is rarely done once, since a model family spans several sizes and products differ in budget and version. Routes that learn from live teachers, such as on-policy distillation, run the experts again whenever a new student is trained. Yet the experts' own RL training has already logged complete trajectories with their outcomes and behavior probabilities. Such a history records not only what an expert can do, but also how it came to do it. We present HARP, a fully offline framework that consolidates reasoning and long-horizon agency from these histories without running the experts again. The historical path curriculum releases experience along the teachers' own training stages, and the success-ratio budget turns successful trajectories into a provable lower bound on task success. A student then holds a non-decreasing floor on this bound for all acquired tasks. Once frozen, the history supports new students without expert inference or student rollouts for optimization. Experiments with Qwen3.5 at three scales show that HARP is ❶ *effective*, recovering 103.6–105.5% of single-task RL gains on same-size pairs and lifting the 9B student of the 27B history from 22.5% to 39.3% on Terminal-Bench 2.1, ❷ *reusable*, training 4B, 9B, and 27B students from one 27B history with zero new expert inference and 52.3% fewer marginal GPU hours than multi-teacher on-policy distillation at matched quality, and ❸ *self-sufficient*, recovering 97.3% of the weakest-domain gain on eight GPUs with the experts offline. Our code and the record schema of teacher histories are available here.

1 INTRODUCTION

Reinforcement learning (RL) with verifiable rewards has become the primary route to capable large language models (LLMs), and it is usually run one domain at a time (Guo et al., 2025), on verified answers for mathematics and code and in executable environments for terminal and web agents (Yu et al., 2025; Gandhi et al., 2026). Deployment, however, calls for a single policy that both reasons correctly and acts persistently. Consolidating several RL experts into one generalist has therefore become a standard final stage of frontier post-training (Team et al., 2025; DeepSeek-AI et al., 2025; Team et al., 2026). Such consolidation is also rarely done once. A model family spans several sizes, and products differ in budget and version, so new students keep being trained from the same expertise (Muralidharan et al., 2024). DeepSeek-R1, for instance, fine-tuned six students from 1.5B to 70B parameters on the samples of a single teacher (Guo et al., 2025).

By what a new student requires, existing reference-based consolidation routes fall into three families (Figure 1, Left). The first is (I) **weight-space merging**, which composes the parameter deltas of the experts without further training (Ilharco et al., 2023; Yadav et al., 2023). While free of gradient updates, it needs element-wise compatible parameters, cannot produce a student of another size, and degrades on the sparse updates of RL-trained experts (Wu et al., 2026a). The second is (II) **on-policy distillation**, in which the student generates and the experts score what it generates (Agarwal et al., 2024; Ma et al., 2026). Although it inherits nearly all of a teacher's capability, the experts run again whenever a new student is trained. Replaying teacher prefixes spares the environment but not the teacher (Liao et al., 2026), and prefix caches cannot score the continuations that a new student

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

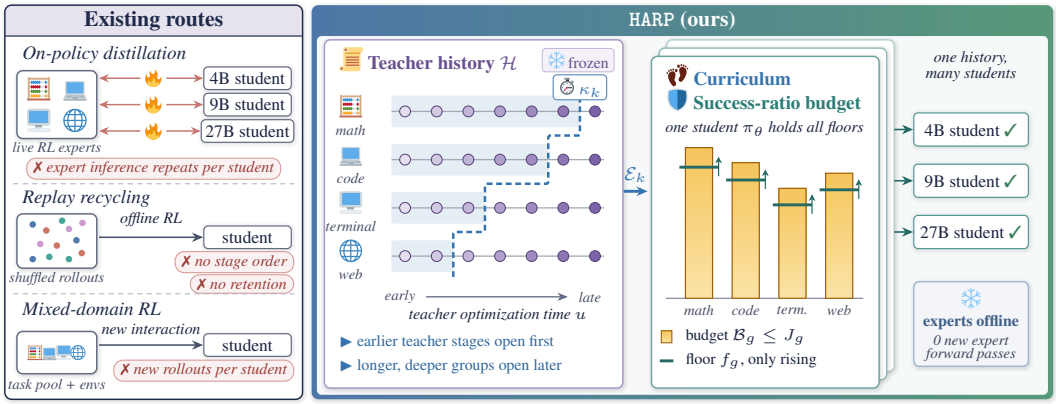


Figure 1: Comparison of HARP with existing consolidation routes.

writes (Kwon et al., 2023). Conversely, the third family is (III) **replay recycling**, which trains the student offline on the rollouts that the experts logged during their own RL training (Chen et al., 2026b). Once the history is recorded, no expert runs again for a new student. A complementary route, mixed-domain RL, retrains the student directly on the pooled tasks with fresh environment interaction (Cheng et al., 2025; Wu et al., 2026b), and we include it as a teacher-free online baseline.

However, reusing a history is not the same as learning from it. Replay recycling treats a teacher's history as an unordered pool of samples, which leaves two gaps open for long-horizon agents whose trajectories span dozens of turns and tens of thousands of tokens. ❶ **Acquisition Gap.** The student meets the converged teacher on its longest interactions from the first update, rather than in the order through which the teacher itself learned them, although a stronger behavior is not necessarily an easier one to absorb (Li et al., 2025). ❷ **Retention Gap.** What the student has acquired is protected, if at all, by a surrogate such as a loss on stored samples or a KL anchor (Lopez-Paz & Ranzato, 2017; Li & Hoiem, 2018), rather than by the success of complete interactions, so it erodes as the student absorbs other experts. Shuffled replay accordingly recovers only 58.8–68.9% of the single-task gain on terminal and web tasks, against 76.6–84.0% on mathematics and code (per-domain G from Tables 1 and 4). Human learning suggests one remedy for both gaps, as a learner absorbs best what lies just beyond its current ability (VYGOTSKY, 1978) and moves on once earlier units are held to criterion (Bloom, 1968). A teacher's RL history carries both ingredients. Its optimization order is a path already walked from the teacher's own initialization, which a same-size student shares, and its successful trajectories, logged with their probabilities, show what holding a capability means. A history thus records not only what an expert can do, but also how it came to do it. This leads us to the following research question:

RQ: *How can one frozen history teach many students without rerunning the experts?*

To this end, we propose HARP (Historical Acquisition and Retention of Policies), a fully offline framework that consolidates capabilities from the RL histories of successful teachers (Figure 1, Right). The **historical path curriculum** decides when a logged trajectory becomes available. It releases the history of a teacher along its own optimization stages, admits longer contexts and deeper feedback progressively, and advances long generations at a finer stride. The **success-ratio budget** decides what is protected once learned. It turns successful trajectories and their behavior probabilities into a provable lower bound on task success, and a checkpoint is accepted only when one student holds the floors of all tasks acquired so far. The **teacher history** is frozen once and shared by all students. A student keeps its own parameters, clock, and floors and pays for its own training. In this way, a new student has to learn again, but the experts do not have to run again. Our contributions are summarized as follows:

- **Problem Identification.** We identify consolidation as a recurring workload in which on-policy routes pay expert inference per student, and show that reusing logged histories removes this cost but leaves acquisition and retention gaps.
- **Practical Solution.** We propose HARP, which releases a frozen teacher history along the teachers' optimization path and holds success floors on a provable lower bound.

- **Reuse Accounting.** We formulate consolidation as a repeated student-production workload, in which one frozen history serves several deployment sizes, amortizes preparation, and removes recurring expert inference.
- **Empirical Validation.** Extensive experiments on five benchmarks show that HARP is ❶ ***effective***, recovering 103.6–105.5% of single-task RL gains on same-size pairs, ❷ ***reusable***, producing three qualified students with 460.0 marginal GPU hours against 963.9 for MOPD, and ❸ ***self-sufficient***, as its students qualify with the experts offline.

2 METHOD

As illustrated in Figure 2, HARP trains students on a frozen teacher history, with the **historical path curriculum** deciding when a record is learned from and the **success-ratio budget** deciding what is protected once learned. The goal is not to stack another learner onto replay, but to let the structure already present in the **teacher history** organize both acquisition and retention.

2.1 PRELIMINARY

Notation. We consider task domains $\mathcal{D}$, here mathematics, code, terminal, and web, whose tasks are partitioned into task groups $g \in \mathcal{G}$. A policy π receives a task $x \sim \rho_g$ and interacts with an environment over turns. At turn t, it reads the context $h_t = C(\mathcal{H}_t)$ that a fixed rule extracts from the environment history $\mathcal{H}_t$, generates an action $a_t \sim \pi(\cdot \mid h_t)$ that holds all tokens of the turn, reasoning and tool arguments included, and receives an observation $o_{t+1} \sim P_g(\cdot \mid \mathcal{H}_t, a_t)$. A trajectory τ of $H(\tau)$ turns is thus distributed as

$$P_\pi^g(\tau) = \rho_g(x) \prod_{t=1}^{H(\tau)} \pi(a_t \mid h_t)\, P_g(o_{t+1} \mid \mathcal{H}_t, a_t). \tag{1}$$

A single-shot answer in mathematics or code is the special case $H = 1$, so reasoning and long-horizon acting fall under one interaction model. A trajectory carries a learning reward $R(\tau)$ and a task outcome $Y(\tau) \in [0, 1]$. The success of π on group g is $J_g(\pi) = \mathbb{E}_{\tau \sim P_\pi^g}[Y(\tau)]$, and L, N, and H denote the context length, the generated tokens, and the feedback turns of a trajectory.

Teacher History. A teacher run e is trained with online RL from the general checkpoint of its size. It follows the policy $\mu_{e,s}$ at optimization step s up to the checkpoint $s_e^\star$ selected on validation. The *teacher history* is the fixed set of records

$$\mathcal{A} = \{(x_i,\ \tau_i,\ R_i,\ Y_i,\ \ell_i^\mu,\ e_i,\ s_i)\}_{i=1}^n, \qquad \ell_i^\mu = \sum_t \log \mu_{e_i,s_i}(a_t \mid h_t), \tag{2}$$

where ℓ_i^μ is the log-probability with which the generating snapshot produced the actor tokens of τ_i, recorded under the actual sampling configuration at generation time. Successes and failures from all stages are kept, and the history is frozen once.

Problem Formalization. A student π_{θ_j}, indexed by $j = 1, \ldots, K$, starts from the general checkpoint of its own size. A same-size student thus shares its teachers' initialization, and a smaller one needs no parameter correspondence. HARP seeks students that maximize $\sum_g J_g(\pi_{\theta_j})$ given only $\mathcal{A}$. Student updates read only this history, so acquisition and retention need no student-generated training trajectories, no new feedback from training environments, and no teacher queries, a setting we call *fully offline*. Registered evaluations still execute the student, and their trajectories stay outside the training records and losses. What the log permits is the likelihood ratio of a complete trajectory,

$$Z_i^{(j)} = \log \frac{P_{\pi_{\theta_j}}^g(\tau_i)}{P_{\mu_i}^g(\tau_i)} = \ell_{\theta_j}(\tau_i) - \ell_i^\mu, \qquad \ell_\theta(\tau_i) = \sum_{n \in \mathrm{actor}(\tau_i)} \log \pi_\theta(y_n \mid h_n), \tag{3}$$

where $\mathrm{actor}(\tau_i)$ indexes the generated tokens y_n of τ_i. The environment terms of Eq. 1 cancel because student and teacher share the task population, the environment, and the context rule, while observations still condition later actions without being scored. We drop j in §2.2 and §2.3.

2.2 HISTORICAL PATH CURRICULUM

The historical path curriculum decides when a logged trajectory becomes available, so that the student meets a behavior when it lies within reach. A strong final teacher need not be the easiest target for a student to absorb. Shuffled replay exposes late and long interactions from the first update, whereas we release them along the teachers' own path, measured in their optimization time and calibrated by the length of what is generated.

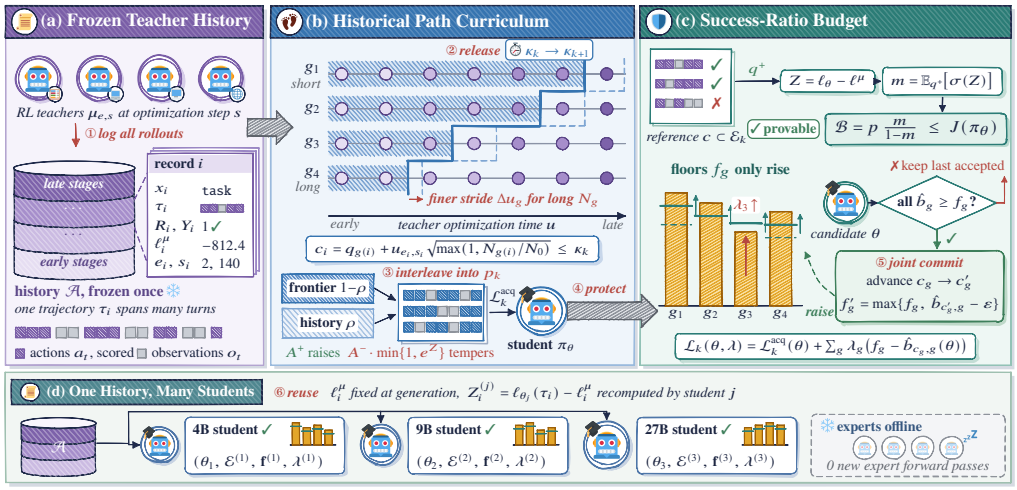


Figure 2: Overview of HARP.

Release Clock. For an accepted run e whose first logged step is s_e^{first}, the normalized optimization time of step s is $u_{e,s} = (s - s_e^{\text{first}})/(s_e^\star - s_e^{\text{first}}) \in [0,1]$, counted in teacher updates rather than episode turns. Tasks are partitioned into a few groups by the context length L_g^{ref} and the feedback depth H_g^{ref} of a fixed reference policy, so that an early failure is not mistaken for an easy task. With within-domain ranks in $[0,1]$, group g enters at level q_g, and record i is released at coordinate c_i,

$$q_g = \max\left\{\text{rank}(L_g^{\text{ref}}),\ \text{rank}(H_g^{\text{ref}})\right\}, \qquad c_i = q_{g(i)} + u_{e_i,s_i}\sqrt{\max\left(1,\ N_{g(i)}/N_0\right)}, \tag{4}$$

where N_g is the generation-length scale of group g and N_0 a common scale. A monotone clock κ_k then opens the released set $\mathcal{E}_k = \{i : c_i \le \kappa_k\}$ at update k. Under this single rule, the earlier teacher stages of a group open first, groups with longer contexts or deeper feedback enter progressively, and a group with longer generations advances along its teachers' path at the finer stride $\Delta u_g = \Delta\kappa/\sqrt{\max(1, N_g/N_0)}$, a calibration that §3 grounds in the tightness of the budget.

Interleaved Sampling. At update k, a domain d is drawn with a fixed weight ω_d, a teacher run uniformly within the domain, and a record from the mixture $p_k = (1-\rho)\,p_{\text{frontier},k} + \rho\,p_{\text{history},k}$ of the newly opened frontier and the history opened before. The earliest groups of all domains take part from the start, so the student does not finish reasoning before it begins to act, and training closes with a joint phase in which the whole history is open. Length thus governs when an experience enters and how fast the path is traversed, rather than how much weight long-horizon tasks receive.

Base Learner. On the released set, the student follows the reward-aware offline objective of replay recycling (Chen et al., 2026b; Le Roux et al., 2025). Let N_i be the number of generated tokens of record i and A_i its advantage, normalized among the released records of the same query, with $A_i^+ = \max(A_i, 0)$ and $A_i^- = \min(A_i, 0)$. The acquisition loss is

$$\mathcal{L}_k^{\text{acq}}(\theta) = -\sum_{d\in\mathcal{D}} \omega_d\, \mathbb{E}_{i\sim p_k(\cdot|d)}\left[\text{sg}\left(A_i^+ + A_i^- \min\{1, e^{Z_i(\theta)}\}\right)\frac{\ell_\theta(\tau_i)}{N_i}\right], \tag{5}$$

where $\text{sg}(\cdot)$ stops the gradient. A positive advantage raises the probability of the behavior, while a negative one is tempered by the truncated ratio, so that an old failure to which the student already assigns little probability is not pushed down further. In this way, the curriculum changes nothing in what the student learns from, only when it learns from it.

2.3 Success-Ratio Budget

The success-ratio budget decides what is protected once learned, namely the task success on acquired groups while the student absorbs other experts. Success, however, is a property of complete interactions that a fully offline learner cannot measure by acting, and surrogates such as a loss on stored samples or a KL anchor constrain likelihood rather than success. We therefore turn the successful trajectories of the history, together with the probabilities under which they were generated, into a lower bound on success that the log alone can compute, and hold this bound above a floor.

Success-Weighted Reference. For group g, a reference c is a fixed panel of logged trajectories from one stage of the history, kept with its original task and attempt weights. It induces a law $P^g_{\mu_c}$ with success $p_{c,g} = J_g(\mu_c) > 0$ and the success-weighted distribution $q^+_{c,g}(\tau) = Y(\tau)\, P^g_{\mu_c}(\tau)/p_{c,g}$, which for binary outcomes is the law of the successful trajectories of the reference.

Budget. Let Z_c be the log-ratio of Eq. 3 taken against the snapshot that generated the panel trajectory. We pass it through the bounded relative ratio $\sigma(Z_c) = w/(1+w)$ with $w = e^{Z_c}$ (Yamada et al., 2011), average over the successful trajectories, and map back to success with an odds transform,

$$m_{c,g}(\theta) = \mathbb{E}_{\tau\sim q^+_{c,g}}\big[\sigma\big(Z_c(\tau;\theta)\big)\big], \qquad \mathcal{B}_{c,g}(\theta) = p_{c,g}\,\frac{m_{c,g}(\theta)}{1-m_{c,g}(\theta)}. \tag{6}$$

Here $m_{c,g}$ is not itself a success rate. The odds transform makes $\mathcal{B}_{c,g}$ a lower bound on the success of the student, $\mathcal{B}_{c,g}(\theta) \le J_g(\pi_\theta)$, with equality at $\pi_\theta = \mu_c$, where $m_{c,g} = \frac{1}{2}$ (Proposition 1).

Constrained Objective. An active group g keeps one reference c_g and one log-floor f_g. With $b_{c,g} = \log \mathcal{B}_{c,g}$, the student minimizes $\mathcal{L}^{\mathrm{acq}}_k(\theta)$ subject to $b_{c_g,g}(\theta) \ge f_g$ for all $g \in \mathcal{G}^{\mathrm{act}}_k$, which we optimize in the primal-dual form

$$\mathcal{L}_k(\theta,\lambda) = \mathcal{L}^{\mathrm{acq}}_k(\theta) + \sum_{g\in\mathcal{G}^{\mathrm{act}}_k} \lambda_g\big(f_g - \hat{b}_{c_g,g}(\theta)\big), \qquad \lambda_g \ge 0, \tag{7}$$

where $\hat{b}$ estimates b on the reference panel, the multipliers ascend on violated constraints within a bounded range, and the student descends. The records of a reference all lie in $\mathcal{E}_k$, so the base update, the query statistics, and the budget share one release boundary, and a later success cannot reach the student early through the protective term. The protective gradient $\nabla_\theta \log \hat{\mathcal{B}} = \sum_i \beta_i \nabla_\theta \ell_\theta(\tau_i)$ is a non-negative weighted sum of trajectory scores with $\sum_i \beta_i \le 1$ (Appendix B), so the implementation avoids multiplying scores by an unbounded full-trajectory importance weight.

Joint Commit. A finite number of primal-dual steps may leave the working student temporarily infeasible, so the accepted state is kept apart from the working iterate. At a commit point, one candidate θ is first checked against all current floors under all current references, and the accepted state stays unchanged if any floor fails. Otherwise, a group may advance its reference to the adjacent released stage c'_g, provided the budget of the candidate under c'_g is no lower than under c_g, and its floor rises to

$$f'_g = \max\big\{f_g,\ \hat{b}_{c'_g,g}(\theta) - \varepsilon\big\}, \tag{8}$$

where ε is a fixed tolerance, and a newly supported group starts at its current budget minus ε. The candidate is thus a common feasible witness of all new constraints. Student, references, active groups, and floors are committed together, a change of reference does not reset a floor, and the tolerance is not granted anew at later stages. In this way, one student holds what the curriculum has built, and the output is the last jointly accepted checkpoint rather than per-task best models.

2.4 One History, Many Students

The history is written once, while students keep arriving as sizes, budgets, and versions change, and a new student reuses it without the experts but still learns by itself.

Per-Student State. The behavior log-probability ℓ^μ_i is fixed at generation time and does not change when the student becomes 4B, 9B, or 27B, since the three sizes share one tokenizer and chat template. Student j only recomputes its own likelihood in Eq. 3 and keeps its own state $(\theta_j,\ \mathcal{E}^{(j)}_k,\ \mathbf{f}^{(j)},\ \lambda^{(j)})$. The clock and references come from one read-only asset, while the student computes and pays for its likelihoods, protective gradients, and floors.

Zero-Expert Condition. Reuse requires that the frozen history holds complete actions and observations, the original rewards and outcomes, correct behavior log-probabilities, and the provenance and grouping of all records. Consolidation then reads only these records and the initial student, with the teacher checkpoints and services disabled, since the budget reads the recorded denominator and advancing a reference switches to other stored records.

3 Theoretical Analysis

We now show why the budget faithfully proxies success, when it is informative, what it guarantees, and what reuse saves, for a fixed group and reference whose subscripts we drop (Appendix C).

Proposition 1 (Success Lower Bound). *For any reference with $p > 0$ and any student π, the budget of Eq. 6 satisfies $0 \le \mathcal{B}(\pi) \le J(\pi)$. Equality holds at $\pi = \mu_c$.*

Proposition 2 (Exact Gap and Local Tightness). *Let $a = \mathbb{E}_{q^+}[w]$ and let $U \ge 0$ be the success mass of the student outside the support of the reference. Then*

$$J(\pi) - \mathcal{B}(\pi) = U + \frac{p}{(1+a)(1-m)}\,\mathbb{E}_{q^+}\Big[\frac{(w-a)^2}{1+w}\Big]. \tag{9}$$

If the teacher path $u \mapsto \mu_u$ is twice differentiable with common support and $S(\tau) = \partial_u \log P_{\mu_u}(\tau)$, the bound built at stage u and evaluated at stage $u+\delta$ satisfies

$$J(\mu_{u+\delta}) - \mathcal{B}_u(\mu_{u+\delta}) = \tfrac{p_u}{2}\,\mathrm{Var}_{q_u^+}(S)\,\delta^2 + o(\delta^2), \qquad p_u\,\mathrm{Var}_{q_u^+}(S) \le C\,N_g, \tag{10}$$

the latter if per-token scores have second moments of at most C and generations at most N_g tokens.

Proposition 2 gives compatibility a precise meaning. Even a strong reference yields a loose bound when the relative probability of the student varies unevenly across its successful trajectories, so an earlier stage close to the student can carry more information than a distant final one, which is what the curriculum supplies. With the stride Δu_g of §2.2, the leading slack of one clock tick is at most $CN_0(\Delta\kappa)^2/2$ for any generation length, so the finer stride calibrates rather than down-weights.

Proposition 3 (Conditional Retention). *Consider the population version of Eqs. 7 and 8. If group g has reached $\mathcal{B}_g(\pi_s) \ge (1-\eta_g)\,J_g(\mu_g^\star)$ after a joint commit s and stays active, then any later accepted student π_t satisfies $J_g(\pi_t) \ge e^{-\varepsilon}\,\mathcal{B}_g(\pi_s) \ge e^{-\varepsilon}(1-\eta_g)\,J_g(\mu_g^\star)$ for $t \ge s$.*

The tolerance ε does not accumulate over later stages or longer trajectories. The guarantee protects what has been acquired, while creating success remains the role of the curriculum. On that side, Lemma 1 (Appendix C) shows that the update of Eq. 5 at the behavior policy is a positive multiple of $\nabla_\theta J$, so an old trajectory near the student points toward improvement.

Proposition 4 (Reuse Accounting). *Let method m pay the history cost C_{hist} and a preparation A_m once, and a marginal cost $c_{m,j}$ for student j, which contains the expert computation $F_{m,j} \ge 0$ left after valid caching. Under the zero-expert condition of §2.4, $F_{\mathrm{HARP},j} = 0$, so the avoided expert computation $I_m(K) = \sum_{j \le K} F_{m,j}$ grows by $F_{m,K+1}$ with a new student, while the net saving $\Delta_m(K) = C_m(K) - C_{\mathrm{HARP}}(K)$ changes by $c_{m,K+1} - c_{\mathrm{HARP},K+1}$.*

Proposition 4 separates what reuse saves from what it costs. Expert computation is saved on any new student, while the net saving can shrink when a new student is cheaper for the comparator, and the fixed costs per student fall as $(C_{\mathrm{hist}} + A_m)/K$.

4 Experiments

In this section, we conduct extensive experiments to answer the following research questions ($\mathcal{RQ}$). ($\mathcal{RQ}$**1**) Is `HARP` effective at consolidating reasoning and agency from old trajectories? ($\mathcal{RQ}$**2**) Do its gains hold across teacher and student scales? ($\mathcal{RQ}$**3**) Can one frozen history serve several students without the experts, and at what cost? ($\mathcal{RQ}$**4**) Why do the curriculum and the budget work?

4.1 Experimental Setup

Evaluation and Benchmarks. Our evaluation covers five benchmarks from four domains, including ❶ *mathematics* on MATH-500 (Hendrycks et al., 2021; Lightman et al., 2024) and AIME25 (Zhang & Math-AI, 2025), ❷ *code* on the 131 tasks of LiveCodeBench v6 in a fixed date window (Jain et al., 2025), ❸ *terminal* tasks on Terminal-Bench 2.1 (Merrill et al., 2026), and ❹ *web* tasks on WA-TG20, a template-isolated test split of 166 WebArena-Verified tasks (hattami et al., 2025). We report the single-attempt success rate, measured at one selected checkpoint per training seed, as the mean and sample standard deviation over three seeds (five for four rows of 9B→4B). The *gain recovery* $G_d = (J_d(\pi) - J_d(\pi_0))/(J_d(\pi_d^{\mathrm{single}}) - J_d(\pi_0))$ measures how much of the single-task RL gain a joint student π keeps in domain d, with mathematics averaging MATH-500 and AIME25, where π_0 is the initial student and π_d^{single} the student trained by `HARP` on d alone from the same source. We report its mean (Mean G) and minimum (Min G) over the four domains, so that a strong domain cannot hide a degraded one, and a value above 100% means that the joint student exceeds its single-task reference.

Teacher Histories. Histories are collected on ❶ mathematics with DAPO-Math-17k (Yu et al., 2025), ❷ code with the TACO and PrimeIntellect portions of DeepCoder (Luo et al., 2025), ❸ terminal with Endless Terminals (Gandhi et al., 2026), and ❹ web with the training templates of WebArena-Verified. Two GRPO teachers (Shao et al., 2024) per domain and size give 24 accepted runs, with training tasks de-duplicated against all evaluation sets.

Baselines. We compare HARP against eight baselines, categorized into four groups. ❶ **Offline replay** shares the access of HARP, including REGEN (Chen et al., 2026b), which replays the pooled history, REGEN-KL, which adds a KL anchor to the initial policy, and Success-SFT, which imitates successful trajectories under our curriculum (Kim & Rush, 2016). ❷ **On-policy distillation** additionally generates new student actions and queries the teachers online, including MOPD (Ma et al., 2026), ReOPD (Liao et al., 2026), and D^3-MOPD, which reallocates domains by the reverse-KL progress of the student (Sun et al., 2026). ❸ **Mixed-domain RL**, denoted Mixed-RL, trains the student with online GRPO on the pooled tasks and needs no teacher (Wu et al., 2026b). ❹ **Weight merging**, namely TIES (Yadav et al., 2023), reads the final teacher weights of same-size pairs.

Implementation Details. We instantiate HARP on Qwen3.5-4B, 9B, and 27B (Qwen Team, 2026) in thinking mode with text-only inputs, with three same-size pairs that test consolidation and three cross-size pairs that add compression, where the 27B history serves 4B, 9B, and 27B students. All trained methods share initialization, decoding, and 320 updates of 64 training records on 48 H100 GPUs (Appendix D).

4.2 Main Results ($\mathcal{RQ}1$ & $\mathcal{RQ}2$)

Table 1 compares HARP with all baselines on the three students of the 27B history, and Table 4 in Appendix E covers the remaining pairs. Obs. ❶–❷ answer $\mathcal{RQ}1$ and Obs. ❸ answers $\mathcal{RQ}2$.

Obs. ❶ HARP combines near-complete gain recovery with broad benchmark advantages. As shown in Tables 1 and 4, HARP leads MOPD in 20 of the 30 pair–benchmark cells and the strongest listed baseline in 19, while online baselines keep advantages on the remaining cells. Under the 27B history, the 4B, 9B, and 27B students recover 98.3%, 104.0%, and 105.3% of single-task gains on average, with weakest-domain recovery of 97.3%, 101.2%, and 100.9%. Across the six pairs, HARP leads REGEN by 27.8–33.7 points in Mean G and keeps the highest Min G on four pairs. D^3-MOPD, which reallocates domains with live teachers, reaches a higher Mean G on five pairs, whereas HARP needs neither fresh generations nor online teachers. These results demonstrate that HARP consolidates reasoning and long-horizon agency effectively from old trajectories.

Obs. ❷ The gains concentrate where feedback runs deep, and exceed those of a stronger single-task learner. HARP leads the strongest baseline in 11 of 12 agentic cells but in 8 of 18 reasoning cells. On the deep web tasks of 9B→4B, it succeeds on 37.8% against 31.6% for REGEN-KL and 23.6% for REGEN, while shallow tasks differ by at most 5.0 points (Appendix E.1). To rule out a merely stronger base learner, we compare the loss from single to joint training under one algorithm. Across D4, D9, D27, and C94, HARP shrinks this loss by 4.4–5.4 points on terminal and 6.6–8.5 points on web tasks, against 0.2–1.9 points on mathematics and code.

Obs. ❸ The gains hold across scales, and a compact student challenges its larger initial counterpart. On the same-size pairs, HARP recovers 103.6%, 105.5%, and 105.3% at 4B, 9B, and 27B, a spread of 1.9 points. Under the 27B history, the web success that a single-task student loses relative to the teacher is 31.4, 15.6, and 6.5 points for the 4B, 9B, and 27B students, whereas the loss from single to joint training is 0.5, -1.3, and -1.4 points. What a small student loses is thus lost mainly while absorbing a distant teacher, not while consolidating (Appendix E.3). Notably, the 4B student of the 9B history exceeds the initial 9B model on both agentic benchmarks, with 30.5% against 22.5% on Terminal-Bench 2.1 and 42.5% against 30.9% on WA-TG20.

4.3 Reuse and Resource Analysis ($\mathcal{RQ}3$)

We freeze the 27B history once and train 4B, 9B, and 27B students in a fixed order, counting a student once it reaches a common target, a Min G of 90% at two consecutive validation checks confirmed on the audit split. Obs. ❹–❺ answer $\mathcal{RQ}3$ with Figure 3 and the ledger of Appendix G.

Obs. ❹ One history serves three students without running the experts again. As shown in Figure 3 (Left), HARP executes no expert forward pass for any student, while the expert computation of MOPD accumulates to 39.7, 97.0, and 171.8 GPU hours and that of ReOPD and D^3-MOPD to 102.2 and 102.3. The students still pay for themselves, and the marginal cost of HARP totals

Table 1: Results (%) of three students of one 27B teacher history, mean and sample standard deviation over three seeds. Gray rows are references, and †, ‡, § mark online teachers, teacher weights, and online environments. Best in **bold**, runner-up underlined.

Method	Reasoning			Agentic		Gain Recovery	
	MATH-500	AIME25	LCB-v6	TB 2.1	WA-TG20	Mean G	Min G
Qwen3.5-27B → Qwen3.5-27B							
Initial student	97.3	88.5	78.8	41.2	41.2	0.0	0.0
Single-task (HARP)	98.8 ±0.2	93.4 ±0.4	86.3 ±0.7	52.4 ±1.1	62.4 ±0.9	100.0	100.0
TIES‡	98.7	92.8	85.3	49.1	56.2	79.0	70.0
REGEN	98.6 ±0.2	92.6 ±0.4	84.9 ±0.8	48.9 ±1.1	55.4 ±0.8	75.3	67.2
REGEN-KL	98.7 ±0.2	93.0 ±0.4	85.6 ±0.7	50.8 ±1.1	59.2 ±0.9	88.2	84.9
Success-SFT	98.6 ±0.2	92.8 ±0.4	85.2 ±0.8	49.8 ±1.1	57.3 ±1.1	81.6	76.0
Mixed-RL§	98.7 ±0.2	93.3 ±0.6	86.2 ±1.0	52.2 ±1.3	61.1 ±1.3	96.8	94.0
ReOPD†	98.7 ±0.2	93.2 ±0.4	85.9 ±0.8	51.8 ±1.3	61.3 ±1.2	95.1	94.4
MOPD†§	**99.1** ±0.4	93.3 ±0.5	87.5 ±1.2	**54.6** ±1.5	62.0 ±0.9	109.1	98.1
D³-MOPD†§	**99.1** ±0.2	93.4 ±0.4	**87.9** ±0.7	54.2 ±0.6	62.7 ±1.1	**110.8**	**101.6**
HARP (Ours)	98.9 ±0.1	**93.5** ±0.4	86.3 ±0.7	53.8 ±1.3	**63.8** ±1.1	105.3	100.9
Qwen3.5-27B → Qwen3.5-9B							
Initial student	96.3	79.8	63.7	22.5	30.9	0.0	0.0
Single-task (HARP)	98.5 ±0.2	88.9 ±0.5	77.5 ±0.8	38.2 ±1.1	53.3 ±0.9	100.0	100.0
REGEN	98.0 ±0.2	87.2 ±0.6	74.7 ±0.8	33.2 ±1.1	46.0 ±1.0	74.2	67.4
REGEN-KL	98.2 ±0.2	87.8 ±0.6	76.0 ±0.8	35.8 ±0.9	50.1 ±1.0	86.9	84.9
Success-SFT	98.2 ±0.2	87.5 ±0.5	75.3 ±0.8	34.5 ±1.1	48.0 ±1.0	80.6	76.2
Mixed-RL§	98.1 ±0.2	87.2 ±0.6	75.3 ±0.9	35.1 ±0.9	49.1 ±0.8	81.9	80.2
ReOPD†	98.3 ±0.2	88.3 ±0.6	76.5 ±0.8	37.3 ±1.3	51.8 ±1.2	93.8	93.1
MOPD†§	98.4 ±0.2	89.9 ±0.6	77.0 ±0.8	37.8 ±1.1	52.7 ±1.2	100.2	96.8
D³-MOPD†§	**98.6** ±0.2	**90.2** ±0.7	77.6 ±0.6	39.0 ±1.0	54.2 ±0.9	**106.0**	100.9
HARP (Ours)	98.5 ±0.2	88.9 ±0.6	**77.7** ±0.9	**39.3** ±1.1	**54.6** ±1.1	104.0	**101.2**
Qwen3.5-27B → Qwen3.5-4B							
Initial student	94.8	69.0	53.4	15.7	20.3	0.0	0.0
Single-task (HARP)	97.1 ±0.2	78.0 ±0.6	64.2 ±0.7	27.1 ±0.9	37.5 ±0.9	100.0	100.0
REGEN	96.5 ±0.2	76.3 ±0.7	62.2 ±0.8	22.7 ±0.9	30.4 ±1.1	70.5	58.8
REGEN-KL	96.8 ±0.2	77.1 ±0.5	63.2 ±0.8	24.5 ±0.9	33.9 ±1.1	83.9	76.9
Success-SFT	96.6 ±0.2	76.7 ±0.6	62.6 ±0.8	23.8 ±0.9	32.1 ±1.1	77.5	68.5
Mixed-RL§	96.9 ±0.5	77.4 ±0.6	64.2 ±0.7	26.1 ±0.8	35.9 ±1.0	93.8	90.7
ReOPD†	96.9 ±0.2	77.7 ±0.6	63.5 ±0.8	26.5 ±0.9	35.8 ±1.0	93.6	90.3
MOPD†§	96.9 ±0.2	77.5 ±0.6	64.8 ±0.7	26.2 ±1.1	36.1 ±1.2	95.9	92.2
D³-MOPD†§	96.9 ±0.2	77.6 ±0.4	**65.2** ±0.8	26.6 ±1.0	36.8 ±0.7	**99.0**	94.9
HARP (Ours)	**97.0** ±0.2	**77.9** ±0.5	64.1 ±0.8	**26.8** ±1.3	**37.0** ±1.1	98.3	**97.3**

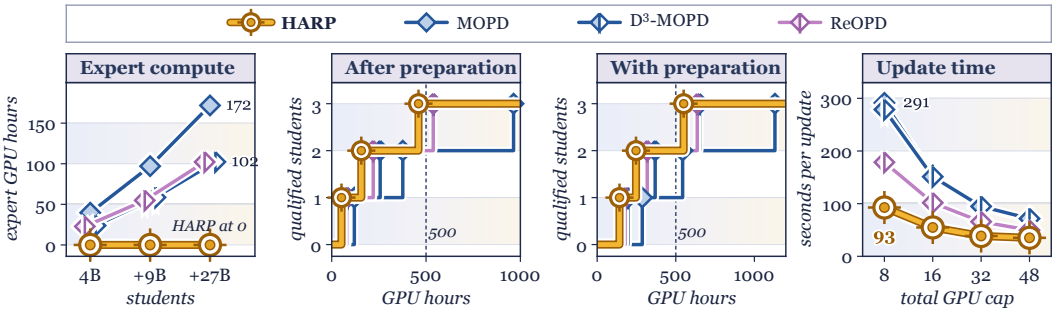


Figure 3: Reuse analysis of HARP with the 27B teacher history.

460.0 GPU hours, against 963.9 for MOPD, 538.1 for ReOPD, and 539.2 for D³-MOPD, a saving of 52.3%, 14.5%, and 14.7%. Notably, the net marginal saving over D³-MOPD falls from 99.9 to 79.2 GPU hours with the 27B student while the avoided expert computation rises, as Proposition 4 anticipates. Offline replay also needs no expert but falls short in quality (Table 1).

Obs. ❺ Students qualify with the experts offline and under device caps. With only the initial student and the read-only history, the teacher weights unmounted and the services unreachable, all three students of HARP finish with zero expert requests, and on eight GPUs the 4B student reaches

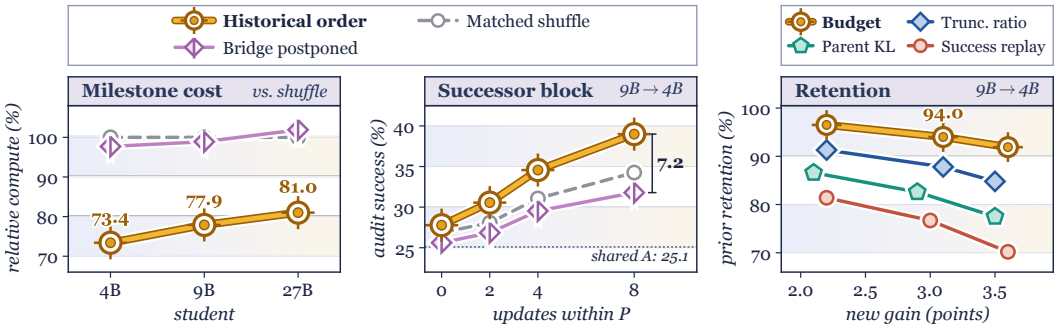


Figure 4: Interventions on the acquisition order (left, middle) and on retention (right).

98.3% Mean G and 97.3% Min G (Table 1). MOPD, ReOPD, and D^3-MOPD need new teacher supervision and cannot run under this access, which we report as outside their information budget rather than as zero success. When the teachers stay available but all services share a cap of 8 GPUs, `HARP` takes 92.7 seconds per update against 291.4 for MOPD and 278.7 for D^3-MOPD (Figure 3, Right). A marginal budget of 500 GPU hours delivers three qualified students with `HARP` and two with the online baselines (Figure 3, After preparation), and `HARP` costs less in 17 of 18 pair-baseline comparisons at the common target.

4.4 Framework Analysis ($\mathcal{RQ}4$)

Ablation Study. We cross the two components with shared learner, rewards, initialization, mixture, and three seeds. ❶ **w/o budget** keeps the curriculum and removes the floors, and ❷ **w/o curriculum** keeps the floors over a shuffled history. On 9B→4B, the variants lift the Mean G of shuffled replay from 71.0% to 85.6% and 89.2%, and the full method reaches 104.4% while raising Min G from at most 85.7% to 100.5%. Replacing the summed log-ratio in the budget by its token mean drops Mean G to 88.3%, about the level without the curriculum, so the gain relies on aggregating over complete trajectories as Proposition 1 requires (Appendix E.2).

Acquisition Order. To isolate timing from content, we switch the budget off and replay one fixed multiset of trajectories under different orders, with identical labels, mixture, exposure, and final segment. The historical order reaches the learning milestone, a validation NLL at 75% of its initial value, with 73.4%, 77.9%, and 81.0% of the compute of matched shuffling at 4B, 9B, and 27B (Figure 4, Left), and on 9B→4B step, length, or reward alone recovers only part of the saving. Postponing the adjacent intermediate stage behind a common successor block P leaves a similar cost (97.7%–101.9%), yet within P the historical order gains 11.2 audit points against 6.2 with the bridge postponed and ends 7.2 points ahead (Figure 4, Middle). It also has the lowest stage-boundary mismatch in all three comparisons, consistent with Proposition 2 (Appendix F.1). With total and final data controlled, timing matters.

Retention under Continued Training. We fork a common parent trained without the budget, continue it under a reasoning-heavy or an agent-heavy mixture, and compare four protective objectives that share parent, references, and extra exposure. On 9B→4B under reasoning pressure, at comparable new gains of 2.9–3.1 points after 48 updates, success replay, a KL anchor to the parent, a truncated trajectory ratio, and our budget keep 76.7%, 82.6%, 87.8%, and 94.0% of the weakest prior gain (Figure 4, Right). Under agent pressure, the budget keeps 95.3% against 83.7% for the KL anchor. The ordering holds on the 9B and 27B students, where the budget keeps 93.2% and 92.1% under reasoning pressure against 82.9% and 83.7% for the KL anchor (Appendix F.2). Constraining success thus preserves more of it than constraining likelihood, at comparable new-task gains.

5 Conclusion

In this paper, we identify consolidation as a recurring workload in which on-policy routes rerun the experts for a new student, while reusing their logged histories leaves acquisition and retention unaddressed. To address both, we propose `HARP`, which releases a frozen teacher history along the teachers' own training path and holds success floors on a provable lower bound. Extensive experiments showcase high recovery of single-task gains across scales, three students of one 27B history at 52.3% lower marginal cost than MOPD without new expert inference, and a 4B student recovering 97.3% of its weakest-domain gain with the experts offline. We believe `HARP` paves the way for treating RL runs as reusable learning structures rather than one-off optimization.

AI USE STATEMENT

In this work, we used generative AI tools to polish the writing of this manuscript. The authors reviewed all edits and take full responsibility for the content of this paper.

ETHICS STATEMENT

This work adheres to the ICLR Code of Ethics. It involves no human subjects, and all training and evaluation tasks are drawn from existing datasets and benchmarks of prior work (§4.1). The terminal and web agents are trained and evaluated only within the sandboxed environments of these benchmarks rather than on live systems. As with other methods that learn from model-generated data, `HARP` may pass undesirable behaviors recorded in a teacher history on to its students, so we recommend auditing a history before reuse and evaluating consolidated agents for safety before deployment. We do not foresee further ethical concerns specific to this work.

REPRODUCIBILITY STATEMENT

The method is specified in §2 and Algorithm 1, with the log-domain evaluation of the budget, the reference panels, and all hyperparameters in Appendix B, and complete proofs in Appendix C. The teacher histories, data splits, evaluation suites, baselines, training configuration, resource accounting, and statistical protocol are listed in Appendix D. We release the code, the analysis scripts that generate all figures and tables, and the record schema of teacher histories, which lists the fields a history logs to be reusable, namely complete trajectories, rewards and outcomes, behavior log-probabilities, the source run and step, and the task group.

REFERENCES

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 22–31. PMLR, 2017. URL http://proceedings.mlr.press/v70/achiam17a.html.

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, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=3zKtaqxLhW.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 41–48, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. doi: 10.1145/1553374.1553380. URL https://doi.org/10.1145/1553374.1553380.

Benjamin S Bloom. Learning for mastery. instruction and curriculum. regional education laboratory for the carolinas and virginia, topical papers and reprints, number 1. *Evaluation comment*, 1(2): n2, 1968.

Tianlei Chen, Jiao Ou, Ziyuan Liu, Ruiming Tang, Jian Liang, and Han Li. Counteraction-aware multi-teacher on-policy distillation for general capability recovery with domain preservation, 2026a. URL https://arxiv.org/abs/2605.27115.

Yunjie Chen, Zihao Chen, Xiaoxin Chen, and Fang Wang. Regen: Replay-recycling for expert-to-generalist distillation with offline reinforcement learning, 2026b. URL https://arxiv.org/abs/2607.19450.

Jorge (Zhoujun) Cheng, Shibo Hao, Tianyang Liu, Fan Zhou, Yutao Xie, Feng Yao, Yuexin Bian, Nilabjo Dey, Yonghao Zhuang, Yuheng Zha, Yi Gu, Kun Zhou, Yuqi Wang, Yuan Li, Richard Fan, Jianshu She, Chengqian Gao, Abulhair Saparov, Taylor W. Killian, Haonan Li, Mikhail

Yurochkin, Eric P. Xing, Zhengzhong Liu, and Zhiting Hu. Revisiting reinforcement learning for LLM reasoning from A cross-domain perspective. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/c2f71567cd53464161cab3336e8fc865-Abstract-Datasets_and_Benchmarks_Track.html.

DeepSeek-AI, Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenhao Xu, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Erhang Li, Fangqi Zhou, Fangyun Lin, Fucong Dai, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoran Wei, Haowei Zhang, Haowen Luo, Haozhe Ji, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, Jialiang Huang, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jingchang Chen, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jinhua Zhu, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexin Huang, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Liang Zhao, Liangsheng Yin, Lihua Guo, Lingxiao Luo, Linwang Ma, Litong Wang, Liyue Zhang, M. S. Di, M. Y Xu, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Panpan Huang, Peixin Cong, Peiyi Wang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, S. H. Liu, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Songyang Zhou, Tao Ni, Tao Yun, Tian Pei, Tian Ye, Tianyuan Yue, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjun Gao, Wentao Zhang, Xi Gao, Xiangwen Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyuan Li, Xu Chen, Xuecheng Su, Xuehai Pan, Xuheng Lin, Xuwei Fu, Y. Q. Wang, Yang Zhang, Yanhong Xu, Yanru Ma, Yao Li, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yiliang Xiong, Ying He, Ying Zhou, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuduan Wang, Yue Gong, Yuhan Wu, Yuheng Zou, Yukun Li, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehua Zhao, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhiyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Zizheng Pan, Zongqing Yao, Bei Feng, Hui Li, J. L. Cai, Jiaqi Ni, Lei Xu, Meng Li, Ning Tian, R. J. Chen, R. L. Jin, S. S. Li, Shuang Zhou, Tianyu Sun, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xinnan Song, Xinyi Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, Dongjie Ji, Jian Liang, Jianzhong Guo, Jin Chen, Leyi Xia, Miaojun Wang, Mingming Li, Peng Zhang, Ruyi Chen, Shangmian Sun, Shaoqing Wu, Shengfeng Ye, T. Wang, W. L. Xiao, Wei An, Xianzu Wang, Xiaowen Sun, Xiaoxiang Wang, Ying Tang, Yukun Zha, Zekai Zhang, Zhe Ju, Zhen Zhang, and Zihua Qu. Deepseek-v3.2: Pushing the frontier of open large language models, 2025. URL https://arxiv.org/abs/2512.02556.

Kanishk Gandhi, Shivam Garg, Noah D. Goodman, and Dimitris Papailiopoulos. Endless terminals: Scaling rl environments for terminal agents, 2026. URL https://arxiv.org/abs/2601.16443.

Alex Graves, Marc G. Bellemare, Jacob Menick, Rémi Munos, and Koray Kavukcuoglu. Automated curriculum learning for neural networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1311–1320. PMLR, 2017. URL http://proceedings.mlr.press/v70/graves17a.html.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhi-

hong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, Hao Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, Tao Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nat.*, 645(8081):633–638, 2025. doi: 10.1038/S41586-025-09422-Z. URL https://doi.org/10.1038/s41586-025-09422-z.

Amine El hattami, Megh Thakkar, Nicolas Chapados, and Christopher Pal. Webarena verified: Reliable evaluation for web agents. In *Workshop on Scaling Environments for Agents*, 2025. URL https://openreview.net/forum?id=94tlGxmqkN.

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 Joaquin Vanschoren and Sai-Kit Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/be83ab3ecd0db773eb2dc1b0a17836a1-Abstract-round2.html.

Gabriel Ilharco, Marco Túlio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=chfJJYC3iL.

Xiao Jin, Baoyun Peng, Yichao Wu, Yu Liu, Jiaheng Liu, Ding Liang, Junjie Yan, and Xiaolin Hu. Knowledge distillation via route constrained optimization. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 1345–1354. IEEE, 2019. doi: 10.1109/ICCV.2019.00143. URL https://doi.org/10.1109/ICCV.2019.00143.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In Jian Su, Xavier Carreras, and Kevin Duh (eds.), *Proceedings of the 2016 Conference on Empirical Methods in*

Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016, pp. 1317–1327. The Association for Computational Linguistics, 2016. doi: 10.18653/V1/D16-1139. URL https://doi.org/10.18653/v1/d16-1139.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114. URL https://www.pnas.org/doi/abs/10.1073/pnas.1611835114.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/0d2b2061826a5df3221116a5085a6052-Abstract.html.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In Jason Flinn, Margo I. Seltzer, Peter Druschel, Antoine Kaufmann, and Jonathan Mace (eds.), *Proceedings of the 29th Symposium on Operating Systems Principles, SOSP 2023, Koblenz, Germany, October 23-26, 2023*, pp. 611–626. ACM, 2023. doi: 10.1145/3600006.3613165. URL https://doi.org/10.1145/3600006.3613165.

Nicolas Le Roux, Marc Bellemare, Jonathan Lebensold, Arnaud Bergeron, Joshua Greaves, Alexandre Fréchette, Carolyne Pelletier, Eric Thibodeau-Laufer, Sándor Tóth, and Sam Work. Tapered off-policy reinforce - stable and efficient reinforcement learning for large language models. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 68152–68180. Curran Associates, Inc., 2025. doi: 10.52202/085713-2289. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/6274d57365d7a6be06e58cad30d1b9da-Paper-Conference.pdf.

Yuetai Li, Xiang Yue, Zhangchen Xu, Fengqing Jiang, Luyao Niu, Bill Yuchen Lin, Bhaskar Ramasubramanian, and Radha Poovendran. Small models struggle to learn from strong reasoners. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, volume ACL 2025 of *Findings of ACL*, pp. 25366–25394. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.FINDINGS-ACL.1301. URL https://doi.org/10.18653/v1/2025.findings-acl.1301.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(12):2935–2947, 2018. doi: 10.1109/TPAMI.2017.2773081. URL https://doi.org/10.1109/TPAMI.2017.2773081.

Baohao Liao, Hanze Dong, Christof Monz, Xinxing Xu, Li Dong, and Furu Wei. Multi-turn on-policy distillation with prefix replay, 2026. URL https://arxiv.org/abs/2607.04763.

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, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Minghuan Liu, Hanye Zhao, Zhengyu Yang, Jian Shen, Weinan Zhang, Li Zhao, and Tie-Yan Liu. Curriculum offline imitating learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 6266–6277. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/31839b036f63806cba3f47b93af8ccb5-Paper.pdf.

Qiang Liu, Lihong Li, Ziyang Tang, and Dengyong Zhou. Breaking the curse of horizon: Infinite-horizon off-policy estimation. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 5361–5371, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/dda04f9d634145a9c68d5dfe53b21272-Abstract.html.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 6467–6476, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/f87522788a2be2d171666752f97ddebb-Abstract.html.

Michael Luo, Sijun Tan, Roy Huang, Ameen Patel, Alpay Ariyak, Qingyang Wu, Xiaoxiang Shi, Rachel Xin, Colin Cai, Maurice Weber, Ce Zhang, Li Erran Li, Raluca Ada Popa, and Ion Stoica. Deepcoder: A fully open-source 14b coder at o3-mini level. https://pretty-radio-b75.notion.site/DeepCoder-A-Fully-Open-Source-14B-Coder-at-O3-mini-Level-1cf81902c14680b3bee5eb349 2025. Notion Blog.

Wenhan Ma, Jianyu Wei, Liang Zhao, Hailin Zhang, Bangjun Xiao, Lei Li, Qibin Yang, Bofei Gao, Yudong Wang, Rang Li, Jinhao Dong, Zhifang Sui, and Fuli Luo. Mopd: Multi-teacher on-policy distillation for capability integration in llm post-training, 2026. URL https://arxiv.org/abs/2606.30406.

Mike Merrill, Alexander Shaw, Nicholas Carlini, Boxuan Li, Harsh Raj, Ivan Bercovich, Lin Shi, Jeong Shin, Thomas Walshe, E. Kelly Buchanan, Junhong Shen, Guanghao Ye, Haowei Lin, Jason Poulos, Maoyu Wang, Marianna Nezhurina, Di Lu, Orfeas Menis Mastromichalakis, Zhiwei Xu, Zizhao Chen, Yue Liu, Robert Zhang, Leon Liangyu Chen, Anurag Kashyap, Jan-Lucas Uslu, Jeffrey Li, Jianbo Wu, Minghao Yan, Song Bian, Vedang Sharma, Ke Sun, Steven Dillmann, Akshay Anand, Andrew Lanpouthakoun, Bardia Koopah, Changran Hu, Etash Guha, Gabriel Dreiman, Jiacheng Zhu, Karl Krauth, Li Zhong, Niklas Muennighoff, Robert Amanfu, Shangyin Tan, Shreyas Pimpalgaonkar, Tushar Aggarwal, Xiangning Lin, Xin Lan, Xuandong Zhao, Yiqing Liang, Yuanli Wang, Zilong (Ryan) Wang, Changzhi Zhou, David Heineman, Hange Liu, Harsh Trivedi, John Yang, Junhong Lin, Manish Shetty, Michael Yang, Nabil Omi, Negin Raoof, Shanda Li, Terry Yue Zhuo, Wuwei Lin, Yiwei Dai, Yuxin Wang, Wenhao Chai, Shang Zhou, Dariush Wahdany, Ziyu She, Jiaming Hu, Zhikang Dong, Yuxuan Zhu, Sasha Cui, Ahson Saiyed, Arinbjörn Kolbeinsson, Christopher Rytting, Ryan Marten, Yixin Wang, Jenia Jitsev, Alex Dimakis, Andy Konwinski, and Ludwig Schmidt. Terminal-bench: Benchmarking agents on hard, realistic tasks in command line interfaces. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 40903–40986, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/444a3737adaee10d86ad2ef5f74468e6-Paper-Conference.pdf.

Mistral-AI, :, Abhinav Rastogi, Albert Q. Jiang, Andy Lo, Gabrielle Berrada, Guillaume Lample, Jason Rute, Joep Barmentlo, Karmesh Yadav, Kartik Khandelwal, Khyathi Raghavi Chandu, Léonard Blier, Lucile Saulnier, Matthieu Dinot, Maxime Darrin, Neha Gupta, Roman Soletskyi, Sagar Vaze, Teven Le Scao, Yihan Wang, Adam Yang, Alexander H. Liu, Alexandre Sablayrolles, Amélie Héliou, Amélie Martin, Andy Ehrenberg, Anmol Agarwal, Antoine Roux, Arthur Darcet, Arthur Mensch, Baptiste Bout, Baptiste Rozière, Baudouin De Monicault, Chris Bamford, Christian Wallenwein, Christophe Renaudin, Clémence Lanfranchi, Darius Dabert, Devon Mizelle, Diego de las Casas, Elliot Chane-Sane, Emilien Fugier, Emma Bou Hanna, Gauthier Delerce, Gauthier Guinet, Georgii Novikov, Guillaume Martin, Himanshu Jaju, Jan Ludziejewski, Jean-Hadrien Chabran, Jean-Malo Delignon, Joachim Studnia, Jonas Amar, Josselin Somerville Roberts, Julien Denize, Karan Saxena, Kush Jain, Lingxiao Zhao, Louis Martin, Luyu Gao, Lélio Renard Lavaud, Marie Pellat, Mathilde Guillaumin, Mathis Felardos, Maximilian Augustin, Mickaël Seznec, Nikhil Raghuraman, Olivier Duchenne, Patricia Wang, Patrick von

Platen, Patryk Saffer, Paul Jacob, Paul Wambergue, Paula Kurylowicz, Pavankumar Reddy Muddireddy, Philomène Chagniot, Pierre Stock, Pravesh Agrawal, Romain Sauvestre, Rémi Delacourt, Sanchit Gandhi, Sandeep Subramanian, Shashwat Dalal, Siddharth Gandhi, Soham Ghosh, Srijan Mishra, Sumukh Aithal, Szymon Antoniak, Thibault Schueller, Thibaut Lavril, Thomas Robert, Thomas Wang, Timothée Lacroix, Valeriia Nemychnikova, Victor Paltz, Virgile Richard, Wen-Ding Li, William Marshall, Xuanyu Zhang, and Yunhao Tang. Magistral, 2025. URL https://arxiv.org/abs/2506.10910.

Saurav Muralidharan, Sharath Turuvekere Sreenivas, Raviraj Joshi, Marcin Chochowski, Mostofa Patwary, Mohammad Shoeybi, Bryan Catanzaro, Jan Kautz, and Pavlo Molchanov. Compact language models via pruning and knowledge distillation. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/4822991365c962105b1b95b1107d30e5-Abstract-Conference.html.

Abhishek Panigrahi, Bingbin Liu, Sadhika Malladi, Andrej Risteski, and Surbhi Goel. Progressive distillation induces an implicit curriculum. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=wPMRwmytZe.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. Rl's razor: Why online reinforcement learning forgets less. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 59839–59864, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/618c95f4557c15b253fb0e6f548ea0c0-Paper-Conference.pdf.

Zechen Sun, Zhiwei Zhang, Fei Zhao, Juntao Li, Mu Chuan, Huayu Deng, Guojian Zhan, Wenliang Chen, Yao Hu, and Min Zhang. D^3-mopd: Dynamic domain scheduling for efficient multi-teacher distillation, 2026. URL https://arxiv.org/abs/2608.24987.

5 Team, Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang Wang, Da Yin, Hao Zeng, Jiajie Zhang, Kedong Wang, Lucen Zhong, Mingdao Liu, Rui Lu, Shulin Cao, Xiaohan Zhang, Xuancheng Huang, Yao Wei, Yean Cheng, Yifan An, Yilin Niu, Yuanhao Wen, Yushi Bai, Zhengxiao Du, Zihan Wang, Zilin Zhu, Bohan Zhang, Bosi Wen, Bowen Wu, Bowen Xu, Can Huang, Casey Zhao, Changpeng Cai, Chao Yu, Chen Li, Chendi Ge, Chenghua Huang, Chenhui Zhang, Chenxi Xu, Chenzheng Zhu, Chuang Li, Congfeng Yin, Daoyan Lin, Dayong Yang, Dazhi Jiang, Ding Ai, Erle Zhu, Fei Wang, Gengzheng Pan, Guo Wang, Hailong Sun, Haitao Li, Haiyang Li, Haiyi Hu, Hanyu Zhang, Hao Peng, Hao Tai, Haoke Zhang, Haoran Wang, Haoyu Yang, He Liu, He Zhao, Hongwei Liu, Hongxi Yan, Huan Liu, Huilong Chen, Ji Li, Jiajing Zhao, Jiamin Ren, Jian Jiao, Jiani Zhao, Jianyang Yan, Jiaqi Wang, Jiayi Gui, Jiayue Zhao, Jie Liu, Jijie Li, Jing Li, Jing Lu, Jingsen Wang, Jingwei Yuan, Jingxuan Li, Jingzhao Du, Jinhua Du, Jinxin Liu, Junkai Zhi, Junli Gao, Ke Wang, Lekang Yang, Liang Xu, Lin Fan, Lindong Wu, Lintao Ding, Lu Wang, Man Zhang, Minghao Li, Minghuan Xu, Mingming Zhao, Mingshu Zhai, Pengfan Du, Qian Dong, Shangde Lei, Shangqing Tu, Shangtong Yang, Shaoyou Lu, Shijie Li, Shuang Li, Shuang-Li, Shuxun Yang, Sibo Yi, Tianshu Yu, Wei Tian, Weihan Wang, Wenbo Yu, Weng Lam Tam, Wenjie Liang, Wentao Liu, Xiao Wang, Xiaohan Jia, Xiaotao Gu, Xiaoying Ling, Xin Wang, Xing Fan, Xingru Pan, Xinyuan Zhang, Xinze Zhang, Xiuqing Fu, Xunkai Zhang, Yabo Xu, Yandong Wu, Yida Lu, Yidong Wang, Yilin Zhou, Yiming Pan, Ying Zhang, Yingli Wang, Yingru Li, Yinpei Su, Yipeng Geng, Yitong Zhu, Yongkun Yang, Yuhang Li, Yuhao Wu, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yuxuan Zhang, Zezhen

Liu, Zhen Yang, Zhengda Zhou, Zhongpei Qiao, Zhuoer Feng, Zhuorui Liu, Zichen Zhang, Zihan Wang, Zijun Yao, Zikang Wang, Ziqiang Liu, Ziwei Chai, Zixuan Li, Zuodong Zhao, Wenguang Chen, Jidong Zhai, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models, 2025. URL https://arxiv.org/abs/2508.06471.

Core Team, Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, Gang Xie, Hailin Zhang, Hanglong Lv, Hanyu Li, Heyu Chen, Hongshen Xu, Houbin Zhang, Huaqiu Liu, Jiangshan Duo, Jianyu Wei, Jiebao Xiao, Jinhao Dong, Jun Shi, Junhao Hu, Kainan Bao, Kang Zhou, Lei Li, Liang Zhao, Linghao Zhang, Peidian Li, Qianli Chen, Shaohui Liu, Shihua Yu, Shijie Cao, Shimao Chen, Shouqiu Yu, Shuo Liu, Tianling Zhou, Weijiang Su, Weikun Wang, Wenhan Ma, Xiangwei Deng, Bohan Mao, Bowen Ye, Can Cai, Chenghua Wang, Chengxuan Zhu, Chong Ma, Chun Chen, Chunan Li, Dawei Zhu, Deshan Xiao, Dong Zhang, Duo Zhang, Fangyue Liu, Feiyu Yang, Fengyuan Shi, Guoan Wang, Hao Tian, Hao Wu, Heng Qu, Hongfei Yi, Hongxu An, Hongyi Guan, Xing Zhang, Yifan Song, Yihan Yan, Yihao Zhao, Yingchun Lai, Yizhao Gao, Yu Cheng, Yuanyuan Tian, Yudong Wang, Zhen Tang, Zhengju Tang, Zhengtao Wen, Zhichao Song, Zhixian Zheng, Zihan Jiang, Jian Wen, Jiarui Sun, Jiawei Li, Jinlong Xue, Jun Xia, Kai Fang, Menghang Zhu, Nuo Chen, Qian Tu, Qihao Zhang, Qiying Wang, Rang Li, Rui Ma, Shaolei Zhang, Shengfan Wang, Shicheng Li, Shuhao Gu, Shuhuai Ren, Sirui Deng, Tao Guo, Tianyang Lu, Weiji Zhuang, Weikang Zhang, Weimin Xiong, Wenshan Huang, Wenyu Yang, Xin Zhang, Xing Yong, Xu Wang, Xueyang Xie, Yilin Jiang, Yixin Yang, Yongzhe He, Yu Tu, Yuanliang Dong, Yuchen Liu, Yue Ma, Yue Yu, Yuxing Xiang, Zhaojun Huang, Zhenru Lin, Zhipeng Xu, Zhiyang Chen, Zhonghua Deng, Zihan Zhang, and Zihao Yue. Mimo-v2-flash technical report, 2026. URL https://arxiv.org/abs/2601.02780.

L. S. VYGOTSKY. *Mind in Society: Development of Higher Psychological Processes*. Harvard University Press, 1978. ISBN 9780674576285. URL http://www.jstor.org/stable/j.ctvjf9vz4.

Jiaqi Wang, Wenhao Zhang, Weijie Shi, Yaliang Li, and James Cheng. Tcod: Exploring temporal curriculum in on-policy distillation for multi-turn autonomous agents, 2026. URL https://arxiv.org/abs/2604.24005.

Chenrui Wu, Zexi Li, Jiajun Bu, Jiangchuan Liu, and Haishuai Wang. Sparsity curse: Understanding RLVR model parameter space from model merging. In Won Kim, Jae-Gil Lee, Kyuseok Shim, James Caverlee, David F. Gleich, Xiangliang Zhang, Qi He, Faisal Farooq, Rómer Rosales, Sayan Ranu, Haixun Wang, Yan Liu, Carl Yang, Leman Akoglu, Nitesh V. Chawla, and Vagelis Papalexakis (eds.), *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, KDD 2026, Jeju Island, Korea, August 9-13, 2026*, pp. 5384–5394. ACM, 2026a. doi: 10.1145/3770855.3817679. URL https://doi.org/10.1145/3770855.3817679.

Siye Wu, Kai Yang, Yuchen Cai, Xin Xu, Peng-Yuan Wang, Jiaxuan Wang, Jiashun Liu, Jiafei Lyu, Yangkun Chen, Saiyong Yang, and Yanghua Xiao. Consolidating rlvr capabilities across domains: A deep dive into fusion paradigms, 2026b. URL https://arxiv.org/abs/2608.27409.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A. Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/1644c9af28ab7916874f6fd6228a9bcf-Abstract-Conference.html.

Makoto Yamada, Taiji Suzuki, Takafumi Kanamori, Hirotaka Hachiya, and Masashi Sugiyama. Relative density-ratio estimation for robust distribution comparison. In John Shawe-Taylor, Richard S. Zemel, Peter L. Bartlett, Fernando C. N. Pereira, and Kilian Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*, pp. 594–602, 2011. URL https://proceedings.neurips.cc/paper/2011/hash/d1f255a373a3cef72e03aa9d980c7eca-Abstract.html.

Zhuolin Yang, Zihan Liu, Yang Chen, Wenliang Dai, Boxin Wang, Sheng-Chieh Lin, Chankyu Lee, Yangyi Chen, Dongfu Jiang, Jiafan He, Renjie Pi, Grace Lam, Nayeon Lee, Alexander Bukharin, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nemotron-cascade 2: Post-training llms with cascade rl and multi-domain on-policy distillation, 2026. URL https://arxiv.org/abs/2603.19220.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: an open-source LLM reinforcement learning system at scale. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/a4277440d50f1f15d2cb4c14f7e0c0d2-Abstract-Conference.html.

Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2025, 2025.

A Related Work

Capability Integration. Frontier post-training increasingly cultivates domain experts with RL and then consolidates them into one model (Team et al., 2025; DeepSeek-AI et al., 2025; Team et al., 2026). By the artifact of the experts that is reused, prior work falls into four routes. ❶ **Weights.** Model merging composes the parameter deltas of the experts without further training (Ilharco et al., 2023; Yadav et al., 2023), yet the sparse updates of RL-trained experts prove fragile under merging (Wu et al., 2026a). ❷ **Tasks and environments.** Mixed-domain RL (Cheng et al., 2025) and cascaded RL (Yang et al., 2026) retrain one policy on all domains, which runs the environments again and lets the domains interfere. A controlled comparison finds that merging, mixed RL, and multi-teacher distillation differ little on average but widely on single benchmarks (Wu et al., 2026b). ❸ **Teachers.** On-policy distillation scores the student's own rollouts with a live teacher (Agarwal et al., 2024), which extends to multiple RL teachers (Ma et al., 2026), to dynamic domain schedules (Sun et al., 2026), and to multi-turn interaction (Liao et al., 2026; Wang et al., 2026). ❹ **Rollouts.** Sequence-level distillation imitates fresh samples of the final experts (Kim & Rush, 2016; Guo et al., 2025), whereas replay recycling trains on the rollouts that the experts logged during their own RL runs (Chen et al., 2026b). Industrial pipelines also fine-tune on such logs while excluding early-stage traces (Mistral-AI et al., 2025). `HARP` falls within replay recycling.

Learning along a Path. A stronger teacher is not necessarily a better one for a small student (Li et al., 2025). Curricula order learning by designed difficulty (Bengio et al., 2009) or by the progress of the learner (Graves et al., 2017). Closest to us, distillation along the teacher's path supervises the student with intermediate teacher checkpoints (Jin et al., 2019), which provably induces an implicit curriculum (Panigrahi et al., 2025), and COIL (Liu et al., 2021) orders a replay buffer by policy similarity. These works show that the path toward a teacher carries information that its endpoint lacks. They run stored checkpoints as live teachers, handle a single teacher on single-turn prediction or control, or infer the order from policy similarity. `HARP` instead reads the order from the logged optimization steps of several RL teachers and calibrates it to context, feedback depth, and generation length, fully offline.

Offline Learning and Retention. Offline RL copes with distribution shift through conservative value estimates (Kumar et al., 2020), and LLM post-training reuses off-policy samples through truncated importance weights (Le Roux et al., 2025). Trajectory-level ratios, however, degrade with the horizon (Liu et al., 2018), which bounded weight transforms temper (Yamada et al., 2011). Our budget applies such a transform to the success-weighted law and holds the resulting lower bound on

Table 2: Comparison with representative works. **Reuses** names the expert artifact that a method consumes (ckpts for checkpoints, logs for logged rollouts). *Fully offline* asks whether integration avoids new student generation (**Rollout**), environment execution (**Env.**), and teacher inference (**Teacher**). **Order** asks whether learning follows the order in which the teacher itself learned, **Retention** whether an explicit mechanism protects what has been acquired, and **Agentic** whether the method is validated on multi-turn agentic tasks, with ✓, ●, and ✗ for full, partial, and absent.

Method	Reuses	Fully offline			History		Agentic
		Rollout	Env.	Teacher	Order	Retention	
Weight space							
TIES (Yadav et al., 2023)	weights	✓	✓	✓	✗	✗	✗
Joint or sequential RL							
Guru (Cheng et al., 2025)	tasks	✗	✗	✓	✗	✗	✗
Cascade 2 (Yang et al., 2026)	tasks, ckpts	✗	✗	✗	●	●	✓
Distillation with teacher inference							
GKD (Agarwal et al., 2024)	teacher	✗	✓	✗	✗	✗	✗
RCO (Jin et al., 2019)	ckpts	✓	✓	✗	✓	✗	✗
MOPD (Ma et al., 2026)	teachers	✗	✗	✗	✗	✗	✓
D^3-MOPD (Sun et al., 2026)	teachers	✗	✗	✗	✗	●	●
TCOD (Wang et al., 2026)	teacher	✗	✗	✗	●	✗	✓
ReOPD (Liao et al., 2026)	teacher, logs	●	✓	✗	✗	✗	✓
Logged data only							
COIL (Liu et al., 2021)	buffer	✓	✓	✓	●	✗	✗
REGEN (Chen et al., 2026b)	RL logs	✓	✓	✓	✗	✗	✗
HARP (Ours)	RL logs	✓	✓	✓	✓	✓	✓

success as a constraint, in the spirit of constrained policy optimization (Achiam et al., 2017). By what is protected, retention methods guard parameters (Kirkpatrick et al., 2017), losses on stored samples (Lopez-Paz & Ranzato, 2017), or output distributions (Li & Hoiem, 2018), or they rely on on-policy data (Shenfeld et al., 2026). Being fully offline, HARP cannot lean on on-policy data, and it protects task success itself through non-decreasing floors on a provable lower bound.

One Expertise, Many Students. A model family is routinely released at several sizes, often by pruning and distilling one large model (Muralidharan et al., 2024), and a single set of reasoning traces has served students from 1.5B to 70B parameters (Guo et al., 2025). On the serving side, paged key-value caches with prefix sharing reuse computation across requests (Kwon et al., 2023). Such caches, however, only reuse what a teacher has already computed, so they cannot score the new continuations of a new student. HARP reuses a learning history rather than stored outputs, and the experts do not run for a new student. Table 2 places representative works on these axes.

Discussion with Closely Related Works. We further discuss the works closest to ours. REGEN (Chen et al., 2026b) shares our data source and offline regime, and it leaves the use of stage information to future work. It replays the pooled history uniformly on single-turn tasks, whereas HARP orders the history, extends it to multi-turn agents, and adds retention. MOPD (Ma et al., 2026) and its industrial deployments (Team et al., 2026) integrate RL teachers on fresh student rollouts with live teachers, both of which HARP removes. ReOPD (Liao et al., 2026) spares the environment by replaying teacher prefixes that also stem from RL training. It still samples student actions and queries the final teacher, and it schedules over the position within a trajectory rather than over the optimization time of the teacher. Magistral (Mistral-AI et al., 2025) fine-tunes a smaller model on logged RL rollouts and excludes those from early steps. Our order intervention (§4.4) suggests that early traces help when they are used at the right time rather than discarded. COIL (Liu et al., 2021) builds a curriculum over a replay buffer for a single continuous-control task, without behavior probabilities or retention. CaMOPD (Chen et al., 2026a) preserves domain capabilities in multi-teacher on-policy distillation by scheduling dedicated updates with live teachers, whereas HARP holds success floors throughout training and from the log alone. D^3-MOPD (Sun et al., 2026) schedules domains by the reverse-KL progress of the student and keeps a floor on the sampling rate of a domain, which protects exposure rather than success and still queries live teachers. Mixed-RL retrains

the student on the pooled tasks without teachers, so it avoids expert inference but needs new environment interaction for a new student. Finally, zero expert inference is shared by all log-only methods, including REGEN and Success-SFT, so we do not count it as a contribution of **HARP** by itself. What **HARP** adds is the quality that makes such reuse worthwhile.

B Implementation Details of HARP

Algorithm 1 **HARP for one student**

Input: frozen history $\mathcal{A}$, initial student π_{θ_0}, clock $\{\kappa_k\}_{k=1}^{T}$, tolerance ε, commit period M

$\theta \leftarrow \theta_0$; accepted state $(\bar{\theta}, \mathbf{c}, \mathbf{f}) \leftarrow (\theta_0, \varnothing, \varnothing)$; $\lambda \leftarrow \mathbf{0}$

for $k = 1, \dots, T$ **do**

Acquisition (§2.2) — $\mathcal{A}$ read-only

$\mathcal{E}_k \leftarrow \{i : c_i \le \kappa_k\}$ with c_i from Eq. 4 — ▷ release clock

draw a batch of complete trajectories from p_k over $\mathcal{E}_k$ — ▷ interleaved sampling

Retention (§2.3) — references within $\mathcal{E}_k$

$\hat{b}_{c_g,g}(\theta) \leftarrow$ Eq. 11 for $g \in \mathcal{G}_k^{\mathrm{act}}$ — ▷ success-ratio budget

$\theta \leftarrow$ one descent step on Eq. 7; $\lambda_g \leftarrow \mathrm{clip}\big(\lambda_g + \eta_\lambda(f_g - \hat{b}_{c_g,g}),\, 0,\, \lambda_{\max}\big)$

Joint commit (§2.3) — every M updates

if $\hat{b}_{c_g,g}(\theta) \ge f_g$ for all $g \in \mathcal{G}_k^{\mathrm{act}}$ **then** — ▷ one student witnesses all floors

$c'_g \leftarrow$ adjacent released stage if $\hat{b}_{c'_g,g}(\theta) \ge \hat{b}_{c_g,g}(\theta)$; $f'_g \leftarrow$ Eq. 8 — ▷ advance

$(\bar{\theta}, \mathbf{c}, \mathbf{f}) \leftarrow (\theta, \mathbf{c}', \mathbf{f}')$ — ▷ committed together

Output: last jointly accepted student $\pi_{\bar{\theta}}$

Algorithm Summary. Algorithm 1 summarizes the training of one student. At update k, the release clock opens the part of the history whose coordinate has been reached, a batch of complete trajectories is drawn by interleaved sampling, and the student takes a primal-dual step. The budgets of all active groups are evaluated on references inside the released set. At a commit point, one candidate is checked against all floors, and only then are references advanced, floors raised, and the accepted state replaced. The order of these operations is part of the method rather than an implementation choice. A second student runs the same loop on the same $\mathcal{A}$ with its own $(\theta, \mathcal{E}_k, \mathbf{f}, \lambda)$.

Log-Domain Evaluation. With normalized panel weights α_i that preserve the original task, run, and attempt population, the budget is evaluated as

$$\hat{p} = \sum_i \alpha_i Y_i, \qquad M_\pm = \sum_i \alpha_i Y_i\, \sigma(\pm Z_i), \qquad \hat{b} = \min\big\{0,\ \log \hat{p} + \log M_+ - \log M_-\big\}, \tag{11}$$

using log-sigmoid and log-sum-exp so that no full-trajectory probability is exponentiated. The cap at zero only prevents an estimate above one from raising a floor further. It does not turn the empirical budget into a confidence certificate.

Gradient of the Log-Budget. Let $q_i = \alpha_i Y_i / \hat{p}$, $v_i = \sigma(Z_i)$, and $m = \sum_i q_i v_i$. Below the cap,

$$\nabla_\theta \log \hat{\mathcal{B}} = \sum_i \beta_i\, \nabla_\theta \ell_\theta(\tau_i), \qquad \beta_i = \frac{q_i\, v_i(1 - v_i)}{m(1-m)}, \qquad \sum_i \beta_i = 1 - \frac{\mathrm{Var}_q(v)}{m(1-m)} \in [0,1]. \tag{12}$$

With the reference fixed, q_i and $\hat{p}$ do not depend on θ, so $\nabla_\theta \log \hat{\mathcal{B}} = \nabla_\theta m / [m(1-m)]$. Here $\nabla_\theta m = \sum_i q_i\, v_i(1-v_i)\, \nabla_\theta \ell_\theta(\tau_i)$, since $\nabla_\theta \sigma(Z_i) = v_i(1-v_i)\nabla_\theta Z_i$ and $\nabla_\theta Z_i = \nabla_\theta \ell_\theta(\tau_i)$. Moreover, $\mathbb{E}_q[v(1-v)] = m(1-m) - \mathrm{Var}_q(v)$ gives the total mass, which $v \in [0,1]$ keeps within $[0,1]$. The protective gradient thus needs no unbounded full-trajectory importance weight, although the norm of the scores and the finite-sample error remain. With parameters fixed and model randomness matched, the weights β_i are computed first and the gradient is then accumulated over micro-batches, so the computation graph of a whole panel is not held in memory at once. The change of reference at a commit is a discrete acceptance step and is not differentiated through.

Mixed-Snapshot References. A reference panel may contain trajectories of several snapshots. We treat the snapshot identity as an auxiliary variable with the same weight under the student and the reference laws. The ratio of a panel trajectory is thus computed against the probability with which it was actually generated, rather than against an unknown mixture marginal. Recorded probabilities correspond to the actual temperature, sampling restrictions, and stopping rule at generation time.

Base Learner. Eq. 5 writes the reward-aware objective of REGEN (Chen et al., 2026b) in terms of the sign of a query-relative advantage. REGEN itself selects the branch by a reward threshold, which coincides with the advantage sign for binary rewards but not in general for dense ones. We therefore store the labels and thresholds of all tasks and apply the same convention to REGEN, REGEN-KL, and `HARP`. A query whose released records contain only successes keeps a non-negative likelihood term, and a query without any success receives no artificial positive. Class balance and advantage statistics only read released records. The base term averages over actor tokens, whereas the ratios inside the budget sum over the complete trajectory.

Reference Panels. A group keeps 32 fixed reference queries with 2 attempts per teacher run, which gives 128 raw records per stage over two runs, successes and failures included. Panels are recorded during teacher training and are not re-sampled once student training starts. A panel with fewer than 16 successes or fewer than 8 distinct successful queries is flagged as weakly supported and reported as such rather than removed.

Table 3: Hyperparameters of `HARP`, shared by all pairs unless stated otherwise.

Component	Setting	Component	Setting
History stages	4	Log tolerance ε	0.10
Task groups per domain	2	Multiplier step size η_λ	10^{-7}
Common length scale N_0	4096 tokens	Multiplier bound $\lambda_{\max}$	10^{-3}
History ratio ρ	0.20	Budget evaluation period	20 updates
Full release of the history	update 224 of 320	Commit period M	40 updates
Domain weights ω_d	uniform	Reference records per stage	128
Single-task updates	80, release at 56	Single-task periods	5 / 10 updates
Parameters / log-softmax	BF16 / FP32	Trajectory sums	FP64

Hyperparameters. Table 3 lists the settings of `HARP`. Multipliers start at zero. The clock opens the whole history at update 224, after which training continues on the whole history for the remaining 96 updates. Short validation trials of 80 updates use a local clock with 4 warm-up updates, full release at update 56, protection every 5 updates, and commits every 10 updates, so that a short trial covers the whole procedure.

C Proofs

Throughout this section we fix a group g and a reference c and drop their subscripts. All trajectory laws share the task population, the environment mechanism, the tool protocol, and the context rule of §2.1. On the support of the reference we write $w(\tau) = P_\pi(\tau)/P_\mu(\tau) = e^{Z(\tau)}$, and we recall $p = J(\mu) > 0$, $q^+(\tau) = Y(\tau)P_\mu(\tau)/p$, $m = \mathbb{E}_{q^+}[w/(1+w)]$, and $\mathcal{B} = p\,m/(1-m)$.

C.1 Proof of Proposition 1

Let $a = \mathbb{E}_{q^+}[w]$. By the definition of q^+,

$$p\,a = \sum_{\tau \in \mathrm{supp}(\mu)} Y(\tau)\,P_\mu(\tau)\,\frac{P_\pi(\tau)}{P_\mu(\tau)} = \sum_{\tau \in \mathrm{supp}(\mu)} Y(\tau)\,P_\pi(\tau) \;\le\; J(\pi), \tag{13}$$

so $p\,a$ is the success mass that the student places inside the support of the reference. The function $\phi(w) = w/(1+w)$ is concave on $w \ge 0$, and Jensen's inequality gives $m = \mathbb{E}_{q^+}[\phi(w)] \le \phi(a) = a/(1+a)$. Since $x \mapsto x/(1-x)$ is increasing on $[0,1)$, it follows that $m/(1-m) \le a$, and therefore

$\mathcal{B} = p\,m/(1-m) \le p\,a \le J(\pi)$. Non-negativity is immediate from $m \in [0,1)$. At $\pi = \mu$, we have $w \equiv 1$, $m = \frac{1}{2}$, and $\mathcal{B} = p = J(\mu)$. More generally, Jensen's inequality is tight when w is constant on the support of q^+, and Eq. 13 is tight when the successful trajectories of the reference cover all the success of the student, which gives the stated equality condition. The argument uses $Y \in [0,1]$ only through q^+ being a probability distribution, so it covers graded outcomes as well as binary ones. □

C.2 Proof of Proposition 2

Exact Gap. Let $U = J(\pi) - p\,a \ge 0$ be the success mass of the student outside the support of the reference (Eq. 13). Then $J(\pi) - \mathcal{B}(\pi) = U + p\,[a - m/(1-m)]$. For any $w \ge 0$ and $a \ge 0$, direct expansion verifies the identity

$$\frac{1}{1+w} = \frac{1}{1+a} - \frac{w-a}{(1+a)^2} + \frac{(w-a)^2}{(1+a)^2(1+w)}. \tag{14}$$

Taking the expectation under q^+, the linear term vanishes since $\mathbb{E}_{q^+}[w] = a$. With $V = \mathbb{E}_{q^+}[(w-a)^2/(1+w)]$, we obtain $1 - m = \frac{1}{1+a} + \frac{V}{(1+a)^2}$, that is, $\frac{a}{1+a} - m = \frac{V}{(1+a)^2}$. Hence

$$a - \frac{m}{1-m} = \frac{a - m(1+a)}{1-m} = \frac{(1+a)\big(\frac{a}{1+a} - m\big)}{1-m} = \frac{V}{(1+a)(1-m)}, \tag{15}$$

which is Eq. 9.

Local Expansion. Let $\pi = \mu_{u+\delta}$ with common support, so that $U = 0$, and let $S = \partial_u \log P_{\mu_u}$. Under twice differentiability and integrability conditions that allow differentiation under the expectation, $w = 1 + \delta S + O(\delta^2)$, $a = 1 + \delta\,\mathbb{E}_{q_u^+}[S] + O(\delta^2)$, and $w - a = \delta\,(S - \mathbb{E}_{q_u^+}[S]) + O(\delta^2)$. The denominator satisfies $(1+a)(1-m)(1+w) \to 2 \cdot \frac{1}{2} \cdot 2 = 2$ as $\delta \to 0$. Substituting into Eq. 9 yields the leading term $\frac{p_u}{2}\mathrm{Var}_{q_u^+}(S)\,\delta^2$ of Eq. 10. If the support changes along the path so that U is not controlled at this order, the expansion does not apply.

Length Bound. Since $Y \le 1$, we have $p_u \mathrm{Var}_{q_u^+}(S) \le p_u \mathbb{E}_{q_u^+}[S^2] = \mathbb{E}_{P_{\mu_u}}[Y S^2] \le \mathbb{E}_{P_{\mu_u}}[S^2]$. The trajectory score is the sum of the per-token scores $S = \sum_n s_n$ with $s_n = \partial_u \log \mu_u(y_n \mid h_n)$, as the environment terms do not depend on u. Under the unconditioned teacher law P_{μ_u}, s_n has zero mean given the past, so $\{s_n\}$ is a martingale difference sequence and the cross terms vanish, $\mathbb{E}_{P_{\mu_u}}[S^2] = \sum_n \mathbb{E}_{P_{\mu_u}}[s_n^2] \le C\,N_g$. The orthogonality is used under P_{μ_u} and not under the success-conditioned law q_u^+, where it need not hold. With the stride of §2.2, $\delta = \Delta\kappa/\sqrt{\max(1, N_g/N_0)}$, and the leading slack is at most $\frac{CN_g}{2} \cdot \frac{(\Delta\kappa)^2}{\max(1, N_g/N_0)} \le \frac{CN_0(\Delta\kappa)^2}{2}$. If N_g is only a scale estimate rather than a hard bound on the generation length, this bound requires a separate check. □

C.3 Proof of Proposition 3

After commit s, Eq. 8 gives $f_g \ge \log \mathcal{B}_g(\pi_s) - \varepsilon$, and floors do not decrease thereafter. Consider a later accepted candidate π_t. Before any change of reference, it satisfies $b_{c_g,g}(\pi_t) \ge f_g$. If the reference advances to c_g', acceptance requires $b_{c_g',g}(\pi_t) \ge b_{c_g,g}(\pi_t) \ge f_g$, and the raised floor $f_g' = \max\{f_g, b_{c_g',g}(\pi_t) - \varepsilon\}$ is met by π_t as well. Proposition 1 holds for any reference, hence $J_g(\pi_t) \ge \mathcal{B}_{c_g^{(t)},g}(\pi_t) \ge e^{f_g^{(t)}} \ge e^{f_g^{(s)}} \ge e^{-\varepsilon}\mathcal{B}_g(\pi_s)$. The second inequality of the proposition is the assumption on $\mathcal{B}_g(\pi_s)$. The tolerance enters once, at the commit that set the floor, and later commits can only raise it. □

C.4 Acquisition Direction of the Base Learner

Lemma 1 (Local Improvement Direction)**.** *Consider a fixed query with binary outcome, a fixed generation length N, and a student equal to the behavior policy μ with $0 < p = J(\mu) < 1$. Under class-balanced sampling with standardized advantages ± 1, the expected ascent direction of Eq. 5 is*

$$g_{\mathrm{acq}} = \frac{1}{2N\,p\,(1-p)}\,\nabla_\theta J(\mu). \tag{16}$$

Proof. Let $s_\theta(\tau) = \nabla_\theta \log P_{\pi_\theta}(\tau)$, $a_+ = \mathbb{E}_{\mu|Y=1}[s_\theta]$, and $a_- = \mathbb{E}_{\mu|Y=0}[s_\theta]$. At $\pi_\theta = \mu$, the truncated ratio equals one, so the two balanced classes contribute $g_{\text{acq}} = \frac{1}{2N}(a_+ - a_-)$. Since $\mathbb{E}_\mu[s_\theta] = p\,a_+ + (1-p)\,a_- = 0$, we have $a_- = -p\,a_+/(1-p)$ and $a_+ - a_- = a_+/(1-p)$. Finally, $\nabla_\theta J(\mu) = p\,a_+$, which gives the claim. □

The coefficient is positive, so an old trajectory acts as a direction of improvement when the student is close to the policy that generated it. The statement is local and does not extend to varying lengths, dense rewards, or a student far from the behavior policy, which is precisely the regime that the curriculum is designed to avoid.

C.5 Proof of Proposition 4

Let K be the number of distinct students, C_{hist} the shared cost of teacher training and history collection, and A_m the one-time preparation of method m, both paid once. For student j, let $U_{m,j}$ be training and protection, $S_{m,j}$ new student rollouts for training, $F_{m,j}$ the expert computation executed after valid caching, $V_{m,j}$ selection and evaluation including their generations, and $L_{m,j}$ log reading, transfer, and other attributable costs, with marginal cost $c_{m,j} = U_{m,j} + S_{m,j} + F_{m,j} + V_{m,j} + L_{m,j}$. In one cost unit,

$$C_m(K) = C_{\text{hist}} + A_m + \sum_{j=1}^{K} c_{m,j}, \qquad \bar{C}_m(K) = \frac{C_{\text{hist}} + A_m}{K} + \frac{1}{K}\sum_{j=1}^{K} c_{m,j}. \tag{17}$$

Under the zero-expert condition of §2.4, Eqs. 5–8 read only the frozen records and the student's own checkpoint, so $F_{\text{HARP},j} = S_{\text{HARP},j} = 0$ for all j, while $U_{\text{HARP},j} > 0$ and $V_{\text{HARP},j} > 0$ in general. Hence $I_{\text{HARP}}(K) = \sum_{j \le K} F_{\text{HARP},j} = 0$, and with $\Delta_m(K) = C_m(K) - C_{\text{HARP}}(K)$,

$$\begin{aligned} \big[I_m(K+1) - I_{\text{HARP}}(K+1)\big] - \big[I_m(K) - I_{\text{HARP}}(K)\big] &= F_{m,K+1} \;\ge\; 0, \\ \Delta_m(K+1) - \Delta_m(K) &= c_{m,K+1} - c_{\text{HARP},K+1}. \end{aligned} \tag{18}$$

The first increment vanishes when all expert requests of student $K+1$ hit the cache, and a linear bound $I_m(K) \ge K\gamma$ follows only if a new student needs at least $\gamma > 0$ of uncached expert computation. The second increment can take either sign, since the saved expert computation may be smaller than the extra protection or I/O of `HARP`. The shared cost C_{hist} cancels in Δ_m, and the fixed cost per student in $\bar{C}_m(K)$ falls as $1/K$. □

D Experimental Details

Models and Pairs. We use the post-trained text policies of Qwen3.5-4B, 9B, and 27B (Qwen Team, 2026) at fixed revisions in thinking mode, without image inputs, low-rank adapters, quantization, or multi-token prediction. The three revisions carry identical tokenizer and chat-template files, so replay scores the same action tokens at any student size. The six pairs are 4B→4B, 9B→9B, 27B→27B, 9B→4B, 27B→9B, and 27B→4B, abbreviated D4, D9, D27, C94, C279, and C274. All pairs with the same teacher size share one accepted history, its behavior probabilities, and its task list. A same-size pair still transfers what domain RL added to the experts, since the student starts from the general checkpoint rather than from any final expert.

Teachers. Teachers are trained with synchronous GRPO (Shao et al., 2024) and 4 rollouts per prompt, with seeds 11 and 23. Math and code use 32 prompts per update for at most 400 updates, and terminal and web use 8 prompts for at most 600 updates. The learning rate is 10^{-6} for 4B and 9B and 5×10^{-7} for 27B, with clipping at 0.2 and a KL coefficient of 0.001 to the initial policy. Validation runs every 50 updates, early stopping is allowed after 100 updates with a patience of 4 evaluations, and the earliest checkpoint within 0.5 points of the best is kept. A run is accepted if it gains at least 2 points on validation with a positive paired lower bound, and agentic runs additionally need at least 10% validation success. The two teachers of a domain carry equal source weight. MOPD and D^3-MOPD route a query group to one teacher in a balanced, seed-fixed alternation, ReOPD scores with the teacher of its prefix, and online teachers and TIES use the selected checkpoint of a run.

Students. A joint student takes 320 updates of 64 training records, 16 per domain, which gives 5,120 records per domain. HARP, REGEN, REGEN-KL, and Success-SFT read complete logged episodes, MOPD, D^3-MOPD, and Mixed-RL generate new episodes, and ReOPD combines replayed prefixes with new actions. D^3-MOPD keeps its adaptive domain allocation within the same 20,480 episodes. A single-task reference takes 80 updates of 64 records under the same normalized clock, with 4 warm-up updates, full release at update 56, protection every 5 updates, and commits every 10, so that protection and commits occur at the same relative progress as in joint training. The learning rate is 10^{-6} for 4B and 5×10^{-7} for 9B and 27B, with AdamW ($\beta = (0.9, 0.95)$, $\epsilon = 10^{-8}$, weight decay 0.01), gradient clipping at 1.0, and a 5% warm-up followed by a constant rate. All methods decode with temperature 1 and top-p 1, without top-k, min-p, or extra penalties. The context limit is 131,072 tokens, math and code generate at most 16,384 tokens, and agentic episodes generate at most 32,768 tokens in total and 4,096 per turn over at most 64 turns, with observations truncated to 4,096 tokens for terminal and 8,192 for web. The response limit of a turn shrinks to the remaining context, and an exhausted context ends the episode, which stays in the denominator and is still scored by the functional verifier.

Data Splits. Code tasks come from the TACO and PrimeIntellect portions of the DeepCoder dataset (Luo et al., 2025). We keep at most 12,000 math, 10,000 code, and 2,000 terminal training tasks, with validation, audit, and internal test splits of 512/128/512 tasks for math and code and 128/128/256 for terminal. Web tasks are grouped by their original intent templates in a fixed release of WebArena-Verified, and template groups are sorted by a SHA-256 hash of a fixed string. The first 113 groups form training, the next 19 validation, the next 20 audit, and the remaining 38 test, with 475, 90, 81, and 166 tasks. An exact Jaccard check over 5-token shingles at a threshold of 0.85 finds no near-duplicate pair across roles, and the other pools are de-duplicated by identifiers, normalized content, and MinHash. Validation serves early stopping and selection, the audit split only evaluates fixed candidates, and the test split is used once for the final results. This rule controls the leakage that our study introduces, but it does not prove that pre-training excluded all public problems.

Evaluation Protocol. MATH-500, AIME25, and LiveCodeBench are attempted 4, 32, and 4 times per task and agentic benchmarks 3 times, giving 2,000, 960, 524, 267, and 498 attempts per checkpoint. We average attempts within a task first and weight tasks equally, which is not a best-of-n score. LiveCodeBench uses release v6 with the half-open window from 2025-02-01 to 2025-05-01, which holds 131 tasks. Terminal-Bench 2.1 uses the revised set of 89 tasks of Terminal-Bench (Merrill et al., 2026), so scores of version 2.0 are not comparable, and WA-TG20 is not the official set of 812 tasks. Terminal policies use a mini-SWE-agent Bash loop that issues shell actions and observes their output and exit status. Web policies use BrowserGym with Playwright and Chromium over accessibility-tree text without screenshots. All conditions share prompts, parsers, observation truncation, resets, and verifiers. Math answers are checked with Math-Verify, code runs in Python 3.11 against the official tests, and terminal and web episodes are judged by functional verifiers. Failures, format errors, timeouts, and exhausted budgets all count in the denominator.

Baselines. REGEN follows the four-domain protocol with the advantage-sign branch of Eq. 5, a code reward equal to the fraction of passed tests, and an outcome Y that requires all tests to pass. REGEN-KL selects its KL coefficient to the initial policy from $\{0.001, 0.01, 0.1\}$ and matches the extra reference exposure of HARP. Success-SFT masks observations and shares our curriculum. MOPD uses the sampled-token log-ratio between teacher and student with clipping at ± 5, and ReOPD samples replayed prefixes with a step decay of 0.6^t. D^3-MOPD updates its domain mixture every 10 updates from the normalized reverse-KL and its descent velocity over three windows of 10 updates, with a mixture temperature of 0.5, a domain floor of 0.10, and batch jitter of 0.30. Mixed-RL trains the initial student with online GRPO on the frozen four-domain fit pools, with 4 responses per query, clipping at 0.2, and a KL coefficient of 0.001, and one set of three runs per student size is reused across teacher-source panels. TIES merges the task vectors of the four domains and two runs, selecting density from $\{0.1, 0.2, 0.5\}$ and scale from $\{0.5, 1.0\}$. It reads the final teacher weights and applies to same-size pairs only. All trained methods receive at most 6 validation trials of 80 updates, developed on 27B→4B for history and teacher methods and on the 4B task pool for Mixed-RL, and the audit and test splits take no part in selection.

Reuse Protocol. The 27B history is frozen once, including tasks, trajectories, rewards, behavior log-probabilities, groups, and reference candidates. Its three students correspond to 27B→4B, 27B→9B, and 27B→27B, and K counts these distinct deployment targets rather than seeds or checkpoints. The cost of a student averages the three common seeds. The one-time preparation A_m sums six development trials and setup, 90.1, 166.9, 98.5, and 112.1 GPU hours for **HARP**, MOPD, ReOPD, and D^3-MOPD, and the history costs 9,323.9 GPU hours once for all methods. A student qualifies when the domain target $T_d = J_{0,d} + 0.9\,(J_d^{\text{single}} - J_{0,d})$ is met in all four domains at two consecutive validation checks and confirmed on the audit split, within four times the cost of REGEN. For MOPD, ReOPD, and D^3-MOPD, we allow prefix key-value caching and exact caches of teacher log-probabilities keyed by the teacher version, the full conditioning sequence, the tokenizer, and the probability rule, and we charge only cache misses. The main order is fixed to 4B, 9B, and 27B, and the other five orders replay the recorded teacher requests from a cold cache. For **HARP**, asset hashes, the model dependency graph, and all teacher-forward, teacher-RPC, and teacher-load events are logged and checked to be zero.

Access and Device Caps. The cluster holds 48 H100 80GB GPUs on six nodes and three CPU servers that host the task environments. The 4B, 9B, and 27B students train on 8, 16, and 32 GPUs with tensor, context, and data parallel degrees of 2/2/2, 4/2/2, and 4/2/4, activation recomputation, and optimizer offloading. GPU cost is the integrated allocated device time, and overlapping services on one device are counted once. We register two access conditions. With logs only, the container holds the initial student and the read-only history, the teacher weights are not mounted, and the teacher services are unreachable. With the experts offline but training environments available, Mixed-RL can also run. In the device-cap setting, a total cap of 8, 16, 32, or 48 GPUs covers the trainer, new rollouts, teacher scoring, protection, and validation. All methods may time-share, offload to CPU, batch automatically, and cache within the cap, and teachers load per domain rather than all at once. The full 8-GPU run of 27B→4B keeps the global batch, the context of 131,072 tokens, and the limit of 64 turns. All four caps, including 8 GPUs, are also profiled from update 224, run 16 native updates with 4 for warm-up and 12 timed, and repeat over three seeds for four methods, which gives 48 profiles of one stage rather than convergence runs.

Statistics. Tables report the sample standard deviation over training seeds, and Initial and TIES use one checkpoint without a seed deviation. Uncertainty is estimated with 2,000 paired bootstrap resamples over training seeds and task clusters, recomputing the initial, single-task, and joint results within a resample. The claim that all domains recover at least 90% of the gain is tested on $J^{\text{joint}} - 0.9\,J^{\text{single}} - 0.1\,J_0$ with simultaneous lower bounds, rather than read from a high mean. When the headroom $J_d^{\text{single}} - J_d(\pi_0)$ of a domain is too small to support a ratio, we report raw success and signed gaps instead of dropping the domain. A value of G_d above 100% means that the joint student exceeds its single-task reference in domain d, not an accuracy above 100%. The number of students K is not treated as a number of independent statistical samples.

Scale of the Study. The study includes 141 main joint-student runs, 24 component-ablation runs, 12 trajectory-ratio ablation runs, 80 single-task runs of **HARP** and 48 of REGEN, 39 order-intervention branches after 9 shared prefixes, 132 continued-training branches, and 15 likelihood probes. Together with 48 device profiles, these form 548 student-training or profiling segments and 80,800 registered updates, where a shared prefix counts its 16 updates once. The 24 teacher runs add at most 12,000 updates and up to 66 development trials add at most 5,280, for at most 638 segments and 98,080 updates on the 48-GPU cluster.

E Additional Results

Remaining Pairs. Table 4 completes the six pairs. On 4B→4B, 9B→9B, and 9B→4B, **HARP** recovers 103.6%, 105.5%, and 104.6%, against 99.7%, 102.6%, and 98.9% for MOPD and 103.7%, 108.3%, and 101.6% for D^3-MOPD. On 9B→4B, the agentic scores of 30.5% and 42.5% exceed those of MOPD by 1.3 and 1.9 points, so the higher recovery is visible in raw success rather than only in the normalization. Mixed-RL, which needs no teacher, recovers 78.4–96.8%, and its recovery varies across panels because one fixed student meets different single-task references.

Table 4: Results (%) on the remaining three teacher→student pairs, in the format of Table 1. On 9B→4B, HARP, REGEN, REGEN-KL, and the single-task reference use five seeds.

Method	Reasoning			Agentic		Gain Recovery	
	MATH-500	AIME25	LCB-v6	TB 2.1	WA-TG20	Mean G	Min G
Qwen3.5-4B → Qwen3.5-4B							
Initial student	94.8	69.0	53.4	15.7	20.3	0.0	0.0
Single-task (HARP)	97.0 ±0.2	77.9 ±0.4	65.1 ±0.6	27.3 ±0.7	38.2 ±0.7	100.0	100.0
TIES‡	96.7	77.3	63.4	24.0	32.5	79.1	68.3
REGEN	96.6 ±0.2	76.5 ±0.3	62.8 ±0.3	23.3 ±0.8	31.9 ±0.9	73.5	64.6
REGEN-KL	96.8 ±0.2	77.0 ±0.3	63.6 ±0.5	25.0 ±0.6	35.0 ±0.6	84.7	79.6
Success-SFT	96.7 ±0.2	76.7 ±0.4	62.9 ±0.6	24.2 ±0.8	33.7 ±0.7	78.9	73.1
Mixed-RL§	96.9 ±0.5	77.4 ±0.6	64.2 ±0.7	26.1 ±0.8	35.9 ±1.0	90.8	86.9
ReOPD†	96.9 ±0.2	77.4 ±0.5	64.5 ±0.6	26.6 ±1.1	36.7 ±0.8	93.6	91.8
MOPD†§	97.4 ±0.4	**78.4** ±0.6	64.7 ±0.7	27.1 ±0.9	37.6 ±0.9	99.7	96.3
D³-MOPD†§	**97.5** ±0.2	78.2 ±0.5	64.8 ±0.8	28.2 ±1.3	38.7 ±0.8	**103.7**	97.3
HARP (Ours)	97.0 ±0.3	77.8 ±0.4	**65.0** ±0.7	**28.6** ±1.1	**39.2** ±0.8	103.6	**99.1**
Qwen3.5-9B → Qwen3.5-9B							
Initial student	96.3	79.8	63.7	22.5	30.9	0.0	0.0
Single-task (HARP)	98.3 ±0.2	87.7 ±0.4	76.1 ±0.7	36.1 ±0.9	50.6 ±0.8	100.0	100.0
TIES‡	98.1	87.2	74.6	31.8	44.2	79.1	67.3
REGEN	97.9 ±0.2	86.5 ±0.5	73.9 ±0.4	31.5 ±1.1	43.8 ±1.0	74.3	65.3
REGEN-KL	98.1 ±0.2	87.1 ±0.5	74.8 ±0.6	33.8 ±0.8	47.6 ±0.8	87.3	83.5
Success-SFT	98.0 ±0.2	86.8 ±0.4	74.3 ±0.6	32.6 ±0.7	45.3 ±0.7	80.3	73.1
Mixed-RL§	98.1 ±0.2	87.2 ±0.6	75.3 ±0.9	35.1 ±0.9	49.1 ±0.8	92.9	92.5
ReOPD†	98.2 ±0.1	87.4 ±0.4	75.4 ±0.9	35.6 ±1.1	49.5 ±0.9	95.4	94.6
MOPD†§	98.2 ±0.1	88.4 ±0.6	77.0 ±1.1	36.0 ±1.4	50.2 ±1.1	102.6	98.0
D³-MOPD†§	98.2 ±0.1	**89.2** ±0.6	**77.2** ±0.7	37.2 ±1.1	50.9 ±0.8	**108.3**	**101.7**
HARP (Ours)	**98.3** ±0.4	87.8 ±0.5	76.3 ±0.9	**37.6** ±1.3	**52.1** ±1.0	105.5	101.2
Qwen3.5-9B → Qwen3.5-4B							
Initial student	94.8	69.0	53.4	15.7	20.3	0.0	0.0
Single-task (HARP)	97.4 ±0.1	79.3 ±0.4	67.0 ±0.5	29.3 ±0.7	40.8 ±0.6	100.0	100.0
REGEN	96.7 ±0.1	77.3 ±0.4	63.8 ±0.6	24.3 ±0.6	33.3 ±0.7	70.9	63.5
REGEN-KL	97.1 ±0.2	78.2 ±0.5	65.2 ±0.6	26.8 ±0.8	37.3 ±0.8	85.1	81.8
Success-SFT	96.9 ±0.2	77.7 ±0.5	64.4 ±0.7	25.6 ±0.9	35.0 ±1.0	77.3	71.6
Mixed-RL§	96.9 ±0.5	77.4 ±0.6	64.2 ±0.7	26.1 ±0.8	35.9 ±1.0	78.4	75.8
ReOPD†	97.2 ±0.2	78.7 ±0.6	66.2 ±1.0	28.3 ±0.9	39.8 ±1.0	94.1	93.0
MOPD†§	**97.6** ±0.2	79.1 ±0.6	66.7 ±0.9	29.2 ±1.0	40.6 ±1.1	98.9	97.7
D³-MOPD†§	**97.6** ±0.1	79.0 ±0.4	67.0 ±0.7	30.1 ±0.6	41.1 ±0.9	101.6	99.4
HARP (Ours)	97.4 ±0.1	**79.3** ±0.5	**67.1** ±0.6	**30.5** ±0.8	**42.5** ±0.8	**104.6**	**100.4**

E.1 Where the Gains Come From

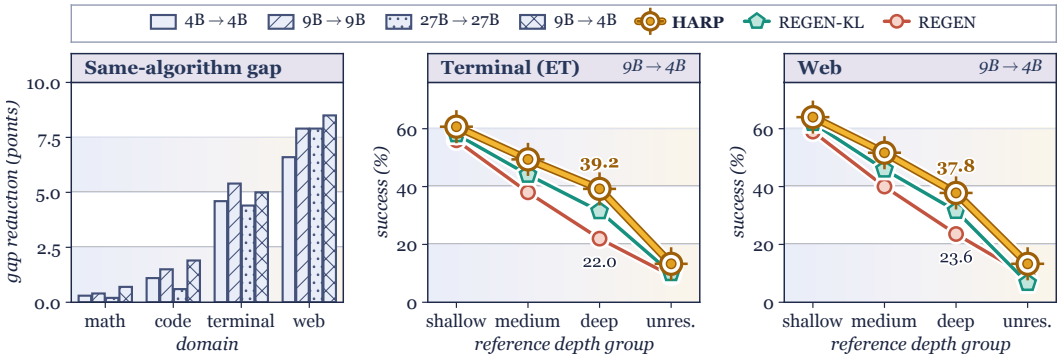


Figure 5: Same-algorithm gap reduction on four pairs and success by feedback depth on 9B→4B.

Same-Algorithm Gap. A joint student may score higher only because its base learner is stronger in isolation. We therefore train single-task students with REGEN as well, under matched per-domain

exposure and three common seeds, and compute the same-algorithm gap reduction

$$D_d = \left(J^{\text{single}}_{\text{REGEN},d} - J^{\text{joint}}_{\text{REGEN},d}\right) - \left(J^{\text{single}}_{\text{HARP},d} - J^{\text{joint}}_{\text{HARP},d}\right), \tag{19}$$

without clipping negative gaps at zero. As shown in Figure 5 (Left), D_d stays within 0.2–1.9 points on mathematics and code but reaches 4.4–5.4 points on terminal and 6.6–8.5 points on web tasks across D4, D9, D27, and C94. This control separates a stronger learner from a smaller consolidation loss, although it does not purely measure gradient interference.

Feedback Depth. We stratify tasks by the feedback depth of a fixed reference, keeping unresolved tasks as a separate group, and use an internal test split of Endless Terminals with 64 tasks per group for terminal tasks. On 9B→4B web tasks, whose 166 tasks fall into groups of 40, 44, 46, and 36, `HARP` succeeds on 64.0%, 51.7%, and 37.8% of the shallow, medium, and deep groups, against 59.0%, 40.0%, and 23.6% for REGEN and 62.0%, 45.9%, and 31.6% for REGEN-KL (Figure 5, Right). Terminal tasks show the same pattern, with 39.2% against 22.0% for REGEN on the deep group. The groups are defined before training and not by student outcomes, and the unresolved group marks tasks that exhaust the turn budget rather than a measured depth. The stratification remains correlational, since depth, difficulty, and generation length can vary together.

E.2 Ablation Study

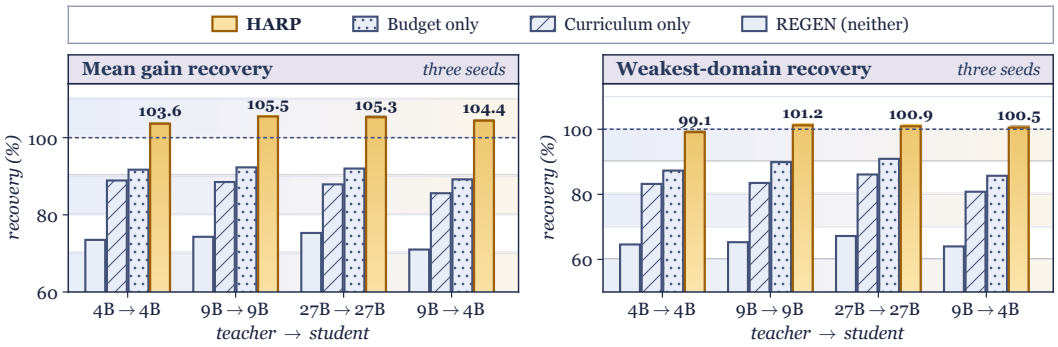


Figure 6: Mean and weakest-domain gain recovery of the four ablation cells on four pairs.

Figure 6 crosses the curriculum and the budget on four pairs with three common seeds, sharing the base learner, actor mask, reward semantics, initialization, and domain mixture. On the three same-size pairs, curriculum only raises the Mean G of REGEN from 73.5–75.3% to 87.9–88.9%, budget only to 91.7–92.3%, and the full method to 103.6–105.5%. The weakest domain moves from 64.6–67.2% for REGEN to 83.2–86.1% and 87.3–90.9% for the single components and to 99.1–101.2% for the full method. The full method scores highest, but we do not infer super-additivity from this alone, which would need a separate estimate of $J_{\text{full}} - J_{\text{curriculum}} - J_{\text{budget}} + J_{\text{shuffle}}$.

Table 5: Full-trajectory against token-mean budget (%), three common seeds. Higher value of the pair in **bold**.

Pair	Budget	MATH-500	AIME25	LCB-v6	TB 2.1	WA-TG20	Mean G	Min G
4B → 4B	Token mean	96.9	77.4	64.4	26.6	35.8	92.2	86.6
	Full (`HARP`)	**97.0**	**77.8**	**65.0**	**28.6**	**39.2**	**103.6**	**99.1**
9B → 9B	Token mean	98.2	87.3	75.4	34.6	48.5	91.7	89.0
	Full (`HARP`)	**98.3**	**87.8**	**76.3**	**37.6**	**52.1**	**105.5**	**101.2**
27B → 27B	Token mean	98.7	93.1	85.7	51.4	60.0	91.4	89.0
	Full (`HARP`)	**98.9**	**93.5**	**86.3**	**53.8**	**63.8**	**105.3**	**100.9**
9B → 4B	Token mean	97.2	78.4	65.9	27.3	37.7	88.3	84.4
	Full (`HARP`)	**97.4**	**79.3**	**67.2**	**30.5**	**42.5**	**104.4**	**100.5**

Full-Trajectory Aggregation. Table 5 replaces the summed actor log-ratio Z in the budget by its token mean Z/N and keeps the curriculum, reference exposure, reference set, multipliers, and commit schedule fixed. The token-mean budget reaches 88.3–92.2% Mean G, about the level of the budget without the curriculum, while the full method reaches 103.6–105.5%. The gain of the full method thus depends on aggregating the log-ratio over complete trajectories, as the lower bound of Proposition 1 requires.

E.3 Scale Analysis

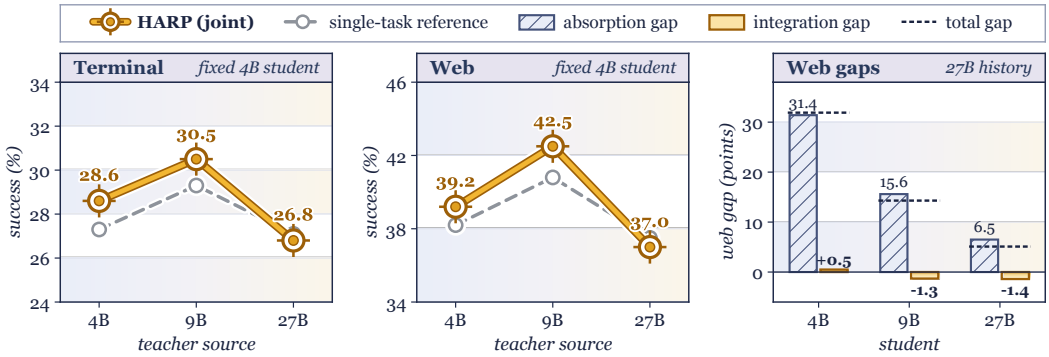


Figure 7: Agentic success of a 4B student across teacher sources and web gap decomposition under the 27B history.

Gap Decomposition. For a fixed task population, the gap between teacher and joint student splits into two signed terms,

$$J_T - J^{\text{joint}} = \underbrace{\left(J_T - J^{\text{single}}\right)}_{\text{absorption gap}} + \underbrace{\left(J^{\text{single}} - J^{\text{joint}}\right)}_{\text{integration gap}}, \tag{20}$$

where the single-task reference comes from the same teacher source. The absorption gap is not an upper bound on what the student can express, and the integration gap is not all of the gradient interference. Under the 27B history (Figure 7, Right), the web absorption gap falls from 31.4 to 15.6 and 6.5 points from the 4B to the 27B student, while the integration gap is 0.5, -1.3, and -1.4 points, for total teacher-to-joint gaps of 31.9, 14.3, and 5.1 points. A negative integration gap means that the joint student exceeds its single-task reference, not that it exceeds the teacher.

Teacher Source. With the student fixed at 4B (Figure 7, Left and Middle), joint terminal success is 28.6%, 30.5%, and 26.8% from 4B, 9B, and 27B teachers, and joint web success is 39.2%, 42.5%, and 37.0%. The adjacent 9B source thus outperforms both the same-size and the largest source. We read this as a question of learnability rather than as evidence that a larger teacher is harmful.

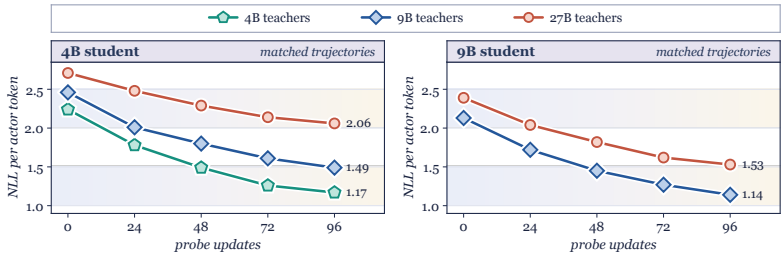


Figure 8: Per-token negative log-likelihood of matched teacher trajectories during 96 probe updates.

Likelihood Probes. To test the learnability reading, we run 15 probes of 96 updates on 4B→4B, 9B→4B, 27B→4B, 9B→9B, and 27B→9B, with three seeds per probe. With the student fixed, the teacher sources share matched queries, balanced for teacher success, reference depth, actor length,

history stage, and run weight. After 96 updates, the per-token negative log-likelihood of a 4B student reaches 1.17, 1.49, and 2.06 nats for the 4B, 9B, and 27B sources, and that of a 9B student 1.14 and 1.53 for the 9B and 27B sources (Figure 8). The lowest negative log-likelihood does not imply the highest task success, since the 9B source still yields the best joint 4B student. The 27B source, however, shows the largest fitting difficulty, consistent with a larger absorption gap. Probe students do not feed back into the main training.

F Mechanism Details

F.1 Acquisition Order

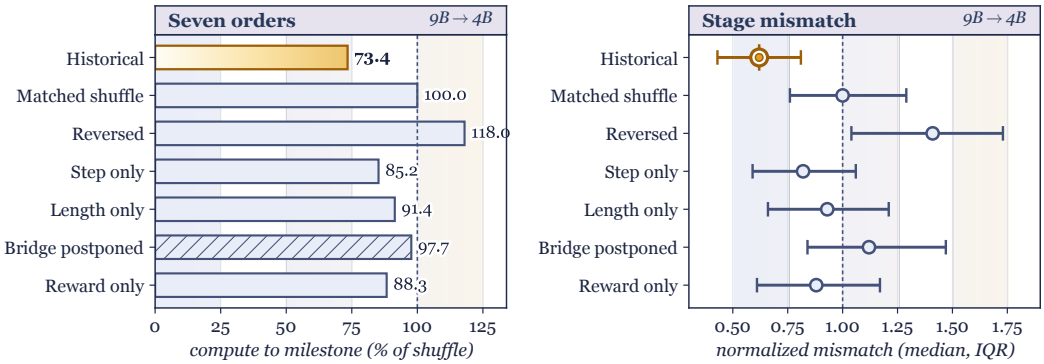


Figure 9: Milestone cost of seven orders and stage-boundary mismatch on 9B→4B.

Protocol. On 9B→4B, we switch off the budget and compare seven orders over 96 updates and three seeds, namely historical, matched shuffle, reversed, step only, length only, bridge postponed, and reward only. The two larger students repeat the historical, matched-shuffle, and bridge-postponed orders, which gives 39 branches in total. All orders share a common prefix A over updates 1–16 and the final 20 updates, and they differ only in the window from update 17 to 76. The prefix of a pair and seed is trained once and forked, so the branches add 80 updates on top of 9 shared prefixes. Within a pair, the orders share the multiset of trajectories, the query-stage packets, the rewards and labels, the domain mixture, and the total exposure. The learning milestone is the first update at which the actor-token NLL of a fixed validation panel of 256 successful teacher trajectories falls to 75% of its initial value, which is 2.480, 2.120, and 1.840 nats on C94, D9, and D27. Ordering by reward alone is a scalar-quality control rather than a reproduction of COIL (Liu et al., 2021).

Bridge Intervention. The bridge test compares the sequences $A \to B \to P \to F \to C$ and $A \to F \to P \to B \to C$ before the shared tail, where B is the adjacent intermediate stage, F a non-adjacent matched block of the same length, P a common successor block at updates 33–40, and C a fixed block. B and F are matched by query, teacher run, reward, and reference depth, with actor tokens differing by at most 5%. P trains on fit logs, while behavior is evaluated on the 81 audit web tasks with four attempts and three seeds. The shared prefix ends at 25.1%, and the historical, matched-shuffle, and bridge-postponed orders enter P at 27.8%, 26.9%, and 25.6%. After 8 updates of P, they reach 39.0%, 34.3%, and 31.8%, so the historical order gains 11.2 points within P against 7.4 and 6.2. Of its 7.2-point lead over the postponed bridge, 2.2 points exist at the start of P and 5.0 points are learned within it.

Results. Figure 9 (Left) shows that the reversed order costs 118.0% of the compute of matched shuffling, while step only, length only, and reward only cost 85.2%, 91.4%, and 88.3%, all above the 73.4% of the historical order. Historical replay has the lowest milestone cost and the lowest stage-boundary mismatch in all three core comparisons, and the reversed order has the highest values on both measures among the seven orders (Figure 9, Right), with median mismatch 0.62 (IQR 0.43–0.81) against 1.00 for shuffling and 1.41 for reversal. Bridge postponement raises mismatch above matched shuffling to 1.12 (IQR 0.84–1.47) while reaching a similar milestone cost, and its lower gain within P provides an additional behavioral comparison. These observations are consistent with

the local compatibility of Proposition 2, although the mismatch alone does not identify it as the only mediator.

F.2 Retention under Continued Training

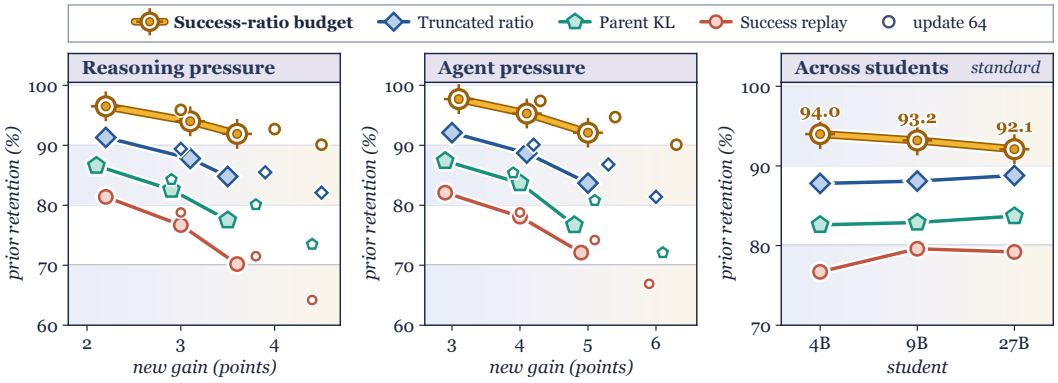


Figure 10: Retention against new gain for four protective objectives.

Protocol. All branches fork from a common curriculum-only student at update 256 and continue for 64 updates. Reasoning pressure weights mathematics, code, terminal, and web by $(0.4, 0.4, 0.1, 0.1)$, and agent pressure reverses the weights. Success replay, a KL anchor to the parent, a truncated trajectory ratio, and the success-ratio budget share the main data, the parent, the reference identities and weights, the extra reference exposure, and the aggregation of actor log-likelihoods. Protection runs every 8 updates and joint acceptance every 16. On 9B→4B, we run four objectives, three strengths, two pressures, and three seeds, plus 12 branches on reference advancement and floor resets. The 9B and 27B students repeat the pre-registered standard strength with 24 branches per student. Audit tasks hold 128 problems for mathematics, code, and terminal and 81 for web, with four attempts and three seeds. Retention is normalized by the common parent rather than by the single-task student,

$$R_d = \frac{J_d(\pi_{\text{post}}) - J_d(\pi_0)}{J_d(\pi_{\text{parent}}) - J_d(\pi_0)}, \tag{21}$$

and we plot the weakest R_d over prior domains against the new gain at continued update 48, with update 64 as the fixed end point. A frozen parent sits at zero new gain, so it cannot win this comparison by forgetting nothing.

Results. Figure 10 shows all three strengths on 9B→4B at update 48, with hollow markers at update 64. Under agent pressure and at the standard strength, the four objectives keep 78.1%, 83.7%, 88.7%, and 95.3% of the weakest prior gain at new gains of 4.0–4.1 points. Under reasoning pressure, the budget keeps 93.2% and 92.1% on the 9B and 27B students at new gains of 2.6 and 1.8 points, against 82.9% and 83.7% for the KL anchor. Since the parents and new gains differ across sizes, we do not connect the three students into one causal frontier.

F.3 Budget Informativeness

Calibration. For the same candidates, a compatible reference yields budgets of 20.0–69.9% against independently measured success of 24.6–74.9%, whereas a distant reference yields only 13.2–54.0% (Figure 11, Left). The success of a student does not depend on the reference, so what changes is how much of it the bound can certify. A compatible reference also keeps the effective sample fraction of the protective gradient at 0.64–0.78, against 0.18–0.31 for a distant one. One weakly supported reference yields a budget of 16.0% above the measured 12.1%, which we report as estimation error rather than clip.

Local Prediction. Proposition 2 predicts a slack close to $p\,\mathrm{Var}_{q^+}(S)\,\delta^2/2$ for small mismatch. On local points, a prediction from 0.2 to 10.4 points matches a measured slack from 0.2 to 11.6 points,

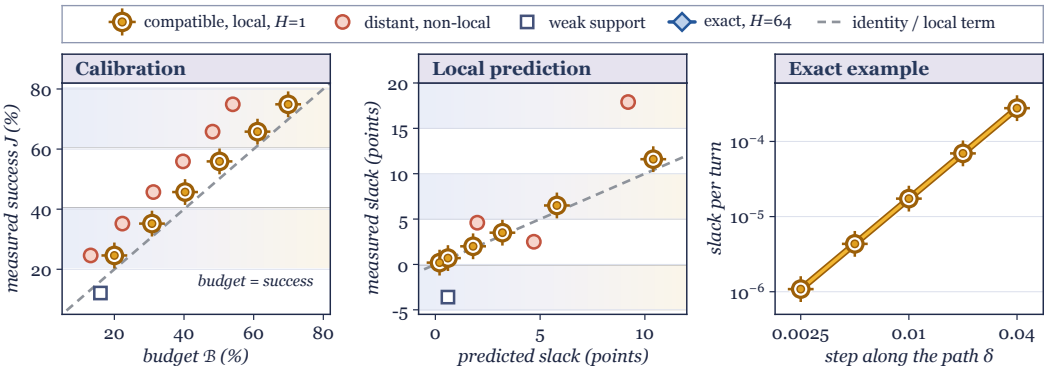


Figure 11: Budget calibration, local slack prediction, and an exactly computable example.

whereas non-local and weakly supported points deviate more and are kept in the plot (Figure 11, Middle).

Exact Example. To check the implementation against the theory, we use a three-outcome example with reference probabilities $(0.3, 0.2, 0.5)$ and outcomes $(1, 1, 0)$, where the student is an exponential tilt along the direction $(0.9, -0.8, 0)$ scaled by $\sqrt{H}$. At $H = 1$ and $\delta = 0.01$, exact summation gives $J = 0.5005586598$ and $\mathcal{B} = 0.5005413211$, a gap of 1.7338772×10^{-5} against a local term of 1.7340×10^{-5}. The normalized local prediction is identical across H, and the exact normalized slack approaches it as the step size tends to zero, while at finite step sizes the exact curves differ by higher-order terms (Figure 11, Right). We further evaluate a finite sequential environment with a binary latent state and an absorbing failure state, where success requires choosing the action of the latent state at all H turns and the state is revealed initially, halfway, or not at all. Exact probability propagation over 30 parameter configurations, $H \in \{1, 4, 16, 64\}$, three information conditions, and five step sizes covers 1,800 configurations. These examples verify arithmetic under given probabilities, and they are neither language-model measurements nor sampled estimates.

G Reuse and Resource Details

Table 6: GPU hours of three quality-matched students of the 27B history, split by cost item. Lowest total per student in **bold**.

Student	Method	Training	Protection	Generation	Expert	Selection	Total
4B	MOPD	23.4	0.0	48.2	39.7	8.6	119.9
	D^3-MOPD	22.1	0.0	30.8	23.9	6.8	83.6
	ReOPD	29.2	0.0	15.3	22.8	11.9	79.2
	HARP	25.7	18.6	0.0	0.0	7.5	**51.8**
9B	MOPD	81.4	0.0	81.9	57.3	36.2	256.8
	D^3-MOPD	58.6	0.0	58.3	34.7	21.6	173.2
	ReOPD	65.7	0.0	23.6	32.1	19.4	140.8
	HARP	51.8	41.7	0.0	0.0	11.6	**105.1**
27B	MOPD	218.7	0.0	204.6	74.8	89.1	587.2
	D^3-MOPD	112.8	0.0	99.6	43.7	26.3	**282.4**
	ReOPD	148.9	0.0	72.4	47.3	49.5	318.1
	HARP	158.1	117.4	0.0	0.0	27.6	303.1
All	MOPD	323.5	0.0	334.7	171.8	133.9	963.9
	D^3-MOPD	193.5	0.0	188.7	102.3	54.7	539.2
	ReOPD	243.8	0.0	111.3	102.2	80.8	538.1
	HARP	235.6	177.7	0.0	0.0	46.7	**460.0**

Per-Student Ledger. Table 6 splits the cost of a student into base training, protection, new student generation, expert forward passes after caching, and selection with evaluation and other attributable

items. **HARP** pays 177.7 GPU hours for protection over the three students, which offline replay does not, and saves the 334.7 and 171.8 GPU hours that MOPD spends on student generation and expert scoring. The history of 9,323.9 GPU hours is counted once for all methods and excluded from these marginal costs. The end-to-end totals of 9,874.0 for **HARP** and 10,454.7 for MOPD thus differ by only 5.6%, and the saving of 52.3% refers to marginal cost.

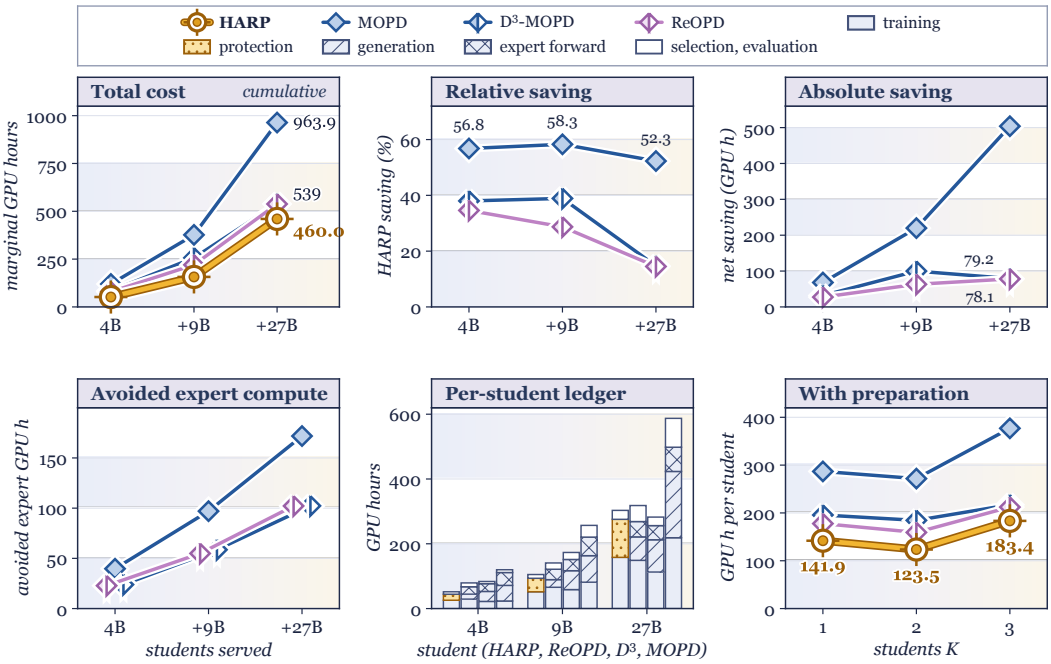


Figure 12: Cost, savings, avoided expert compute, and per-student cost of three students of the 27B history.

Savings Need Not Grow. Figure 12 shows that the total cost of all methods grows with the number of students, since protection and evaluation are paid again for a new student. The relative saving of **HARP** over MOPD moves from 56.8% after the 4B student to 58.3% and then 52.3%, while the avoided expert computation rises from 39.7 to 171.8 GPU hours. Against D^3-MOPD, the net marginal saving falls from 99.9 to 79.2 GPU hours with the 27B student, which D^3-MOPD trains more cheaply, although the avoided expert computation still grows to 102.3. With preparation included, three students cost 550.1, 1,130.8, 636.6, and 651.3 GPU hours for **HARP**, MOPD, ReOPD, and D^3-MOPD, and the fixed cost of **HARP** per student falls from 9,414.0 to 3,138.0 GPU hours as K grows from one to three. Cold-cache replays of the other five orders move the cumulative expert cost of MOPD only within 171.6–172.7 GPU hours.

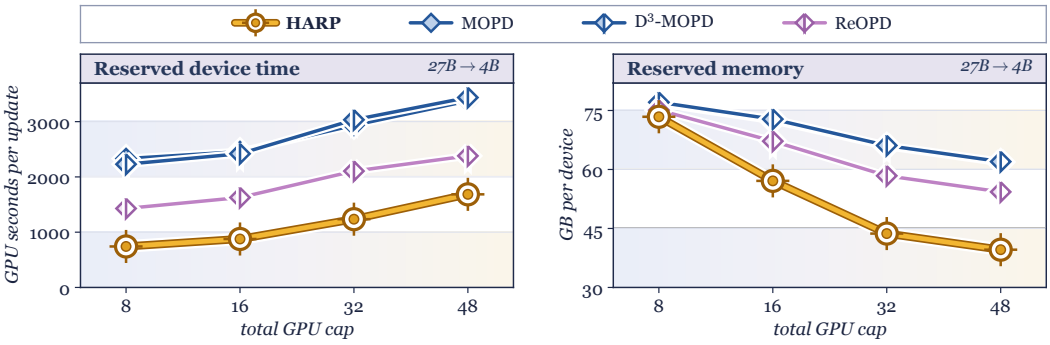


Figure 13: Reserved device time and peak memory per update under a total GPU cap on 27B→4B.

Device Caps. Figure 13 complements the update time of Figure 3. From 8 to 48 GPUs, the reserved device time per update grows from 741.6 to 1,684.8 GPU seconds for HARP and from 2,331.2 to 3,374.4 for MOPD, so more devices finish faster but reserve more device time. The maximum reserved memory per device falls from 73.4 to 39.6 GB for HARP and from 76.8 to 61.8 GB for MOPD. These profiles restart from one registered state of different policies, so the per-update time of two methods is not a causal comparison of teacher overhead, and they are not multiplied by 320 to estimate a full run.

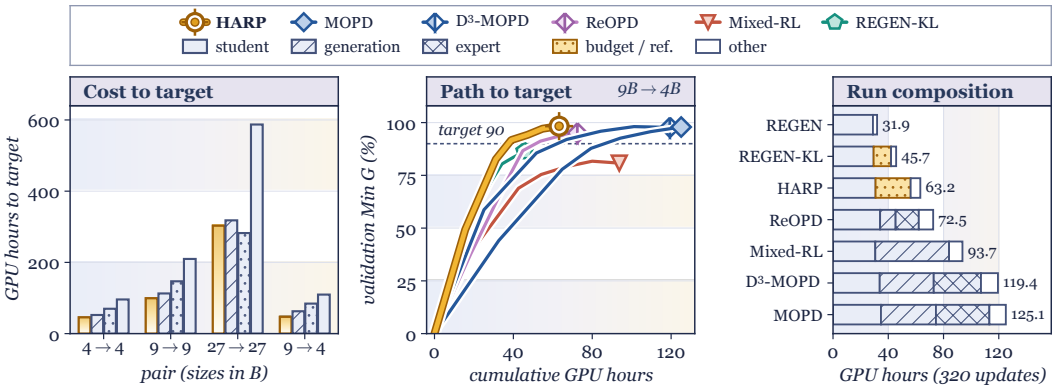


Figure 14: GPU hours to a common quality target for one student, the path to the target, and the composition of run compute.

Cost to One Qualified Student. Before reuse, we compare the cost of one student to reach the common target of Appendix D. HARP reaches it with 45.8, 99.4, 303.1, and 47.2 GPU hours on 4B→4B, 9B→9B, 27B→27B, and 9B→4B, against 95.9, 209.7, 587.2, and 109.6 for MOPD, 52.3, 112.8, 318.1, and 62.8 for ReOPD, and 69.8, 146.9, 282.4, and 84.3 for D^3-MOPD (Figure 14, Left). Over the six pairs and three online baselines, HARP costs less in 17 of 18 comparisons, the exception being D^3-MOPD on 27B→27B. Mixed-RL spends 93.7, 194.6, and 528.9 GPU hours at 4B, 9B, and 27B over 320 updates and reaches the target of the 27B history for 4B and 27B but not for 9B. Protection is not free, as budget evaluation and reference recomputation take 40.2% of the full 9B→4B run of HARP (Figure 14, Right). At 320 updates, the same-size HARP runs cost 59.6, 127.6, and 371.8 GPU hours at 4B, 9B, and 27B, compared with 31.0, 70.6, and 214.6 for REGEN and 111.4, 209.7, and 587.2 for MOPD, an overhead over REGEN of 92.3%, 80.7%, and 73.3%.

G.1 Execution Controls

Execution Budgets. We rerun the three offline methods on 9B→4B, 9B→9B, and 27B→27B under turn caps of 16, 32, and 64, charging all attempts, including failures, to the calls per solved task. On 9B→4B web tasks with 64 turns, HARP succeeds on 42.5% with 53.5 calls per solved task, against 33.3% and 83.8 calls for REGEN and 37.3% and 68.2 calls for REGEN-KL (Figure 15). A lower cap is run independently rather than read from a prefix of a longer run, and a zero success rate leaves the cost undefined rather than zero.

Output Controls. A transient fault rejects one request before the first valid action without changing the environment. Averaging the terminal and web changes equally, HARP changes by −2.9, −2.5, and −2.0 points on 9B→4B, 9B→9B, and 27B→27B, against −5.8, −4.8, and −3.9 for REGEN-KL and −8.7, −6.9, and −5.8 for REGEN (Figure 16, Left). The last jointly accepted checkpoint exceeds the fixed checkpoint at update 320 by only 0.2, 0.1, 0.3, and 0.2 points in agentic success on D4, D9, D27, and C94 (Middle), so the gains do not stem from stopping at a favorable moment. Floors do not decrease by construction, which alone says nothing about actual success. At continued update 48, a variant that resets the floors keeps 76.2% of the weakest prior gain against 95.3% for the full method (Right), and the update-64 results of the four protective objectives appear in Figure 10.

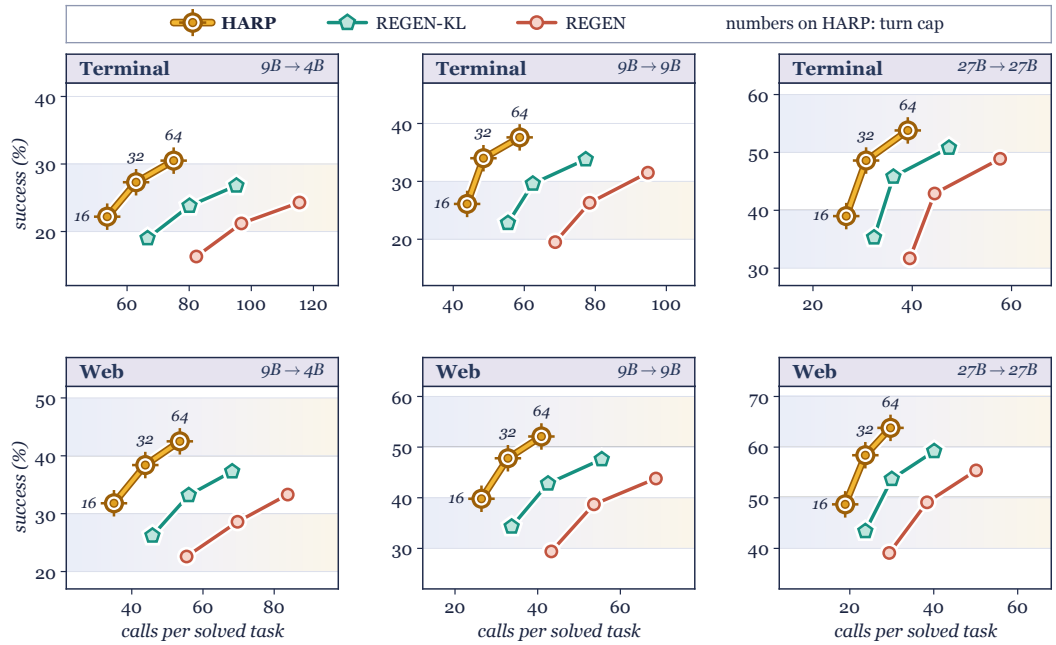


Figure 15: Agentic success against tool calls per solved task under turn caps of 16, 32, and 64.

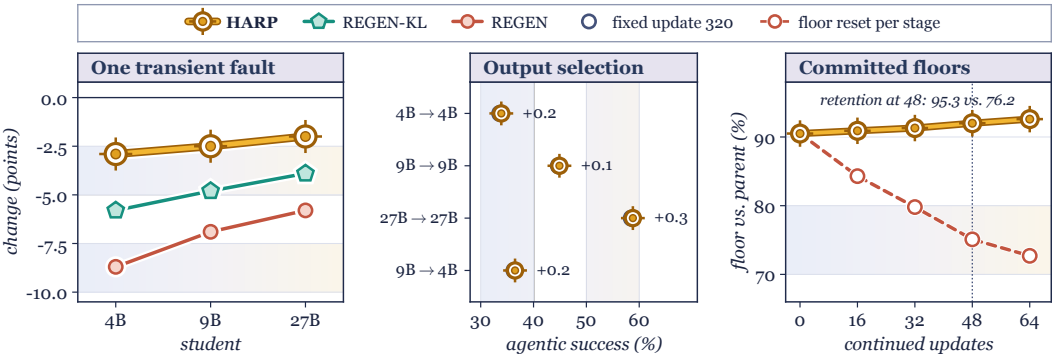


Figure 16: Success change under one transient tool fault, last accepted versus fixed final student, and committed floors.

H Scope of Claims

What Zero Expert Inference Means. Once a complete history has been recorded, consolidation adds no forward pass of a domain expert. This does not make teacher training free, and it does not remove student training, protection, or evaluation, which a new student pays again. If a history lacks the required probabilities, the teacher computation that fills them belongs to asset preparation, and the result is reported as reuse after backfill. Zero expert inference is also shared by REGEN, Success-SFT, and other log-only methods, so it is a common basis of offline consolidation rather than a contribution of `HARP` itself.

Positive Transfer. A Mean G above 100% indicates that the joint student exceeds its single-task references in some domains. Proposition 3 allows but does not predict such transfer, since the budget protects what has been acquired and does not cap later success. The single-task reference is a set of models under the current algorithm and exposure, not an upper bound on capacity.

Bounds and Certificates. The lower bound of Proposition 1 holds for population quantities. Its estimate on a finite, repeatedly used reference panel is an optimization tool rather than an independent certificate, and bounded transforms do not create coverage or remove all horizon-related

estimation error (Liu et al., 2018; Yamada et al., 2011). The floors protect the registered groups and do not extend to new environments or changed tool semantics, where the cancellation behind Eq. 3 requires a new check.

Coverage. Our results cover one model family at three scales, six teacher→student pairs, and three students of one history. They do not form a scaling law and do not extend by themselves to incompatible tokenizers, other architectures, tasks outside the support of the history, or an unbounded number of product versions. New interaction beyond the fixed history may cover missing behavior, but it changes the information budget of this study. Confidence-aware budgets and adaptive advancement of the history under the same offline constraint remain directions for future work.