LEAP: LEARNING FROM EXPERTS' ACQUIRED PROFICIENCY IN MULTI-TEACHER ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-teacher on-policy distillation offers a promising approach to integrating specialized capabilities into a single language model. However, matching each teacher's output distribution may introduce competing supervision and interfere with capabilities acquired from other experts. We introduce LEAP, a framework that focuses on transferring the proficiency experts acquire during post-training. LEAP uses differences between each expert and its pre-specialization reference model to represent acquired proficiency, and integrates these signals on student-generated trajectories. We investigate this approach through supervision analysis, sequential expert distillation, and domain-routed training with varying query mixtures. On matched student rollouts, LEAP reduces conflicting supervision compared with conventional on-policy distillation. In sequential training, it improves retention of previously acquired capabilities while maintaining the acquisition of new skills. Across different query mixtures, LEAP achieves a more favorable trade-off between improving emphasized domains and preserving capabilities in underrepresented domains. These findings support transferring experts' post-training gains as an effective principle for capability integration, enabling more reliable accumulation of specialized expertise in a single student.

1 INTRODUCTION

Recent advances in reinforcement learning for large language models (LLMs) have produced a growing collection of domain-specialized expert models in mathematics, science, coding, and other knowledge-intensive domains. These experts are obtained by post-training a common pretrained model toward different capabilities. This raises a natural question: *how can the complementary capabilities of multiple specialized experts be consolidated into a single compact model?*

On-policy distillation (OPD) provides a natural framework for such capability transfer. OPD trains the student on its own generated trajectories while using a stronger teacher to supervise the student's on-policy distribution. A standard OPD objective can be written as

$$\mathcal{L}_{\mathrm{OPD}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\theta(\cdot|x)} \left[D_{\mathrm{KL}} \left(\pi_\theta(y \mid x) \,\|\, \pi^\star(y \mid x) \right) \right], \tag{1}$$

where π_θ denotes the student policy and $\pi^\star$ the teacher policy. At the token level, the supervision is often simplified as the teacher-student log-probability difference on student-sampled tokens.

While effective for single-teacher transfer, extending OPD to multiple specialized teachers introduces a fundamental difficulty: their supervision signals are not necessarily compatible. Different teachers can assign opposite update directions to the same student-generated token, leading to gradient interference and causing one expert's supervision to partially overwrite capabilities transferred from another.

We find that such conflicts do not necessarily reflect genuine disagreement between expert capabilities. The standard OPD signal compares each teacher directly against the current student, and therefore conflates two effects: the capability acquired by the teacher during specialization and the broader distributional gap between the teacher and the student. When the latter dominates, different teachers may appear to disagree simply because they are independently correcting the same student–base mismatch. We refer to this phenomenon as *spurious teacher conflict*.

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.

Importantly, the training pipeline of multi-teacher OPD naturally provides a useful reference for disentangling these effects. A common paradigm is to start from one or a small number of pretrained base models, specialize them into domain experts through post-training, and then distill these experts into students of different sizes. Consequently, both the post-trained teacher checkpoints and their corresponding base checkpoints are typically available. This structure allows us to explicitly separate what an expert has *acquired during specialization* from what it already inherited from its base model, without introducing an additional reference model or requiring extra training.

Therefore, we introduce **LEAP**, a multi-teacher on-policy distillation framework that transfers the *learned increments* of expert teachers rather than their full predictive distributions. For each teacher T_i, LEAP measures its change relative to the corresponding pre-specialization base model B_i,

$$\Delta_i(y_t) = \log p_{T_i}(y_t \mid x, y_{<t}) - \log p_{B_i}(y_t \mid x, y_{<t}), \tag{2}$$

and uses a consistency gate to suppress updates that conflict with the student's on-policy learning direction. By centering each teacher against its own base model, LEAP removes much of the shared pretrained component and isolates the changes introduced during expert post-training. These base-centered deltas provide a cleaner representation of specialized capability increments and are therefore more suitable for multi-expert composition.

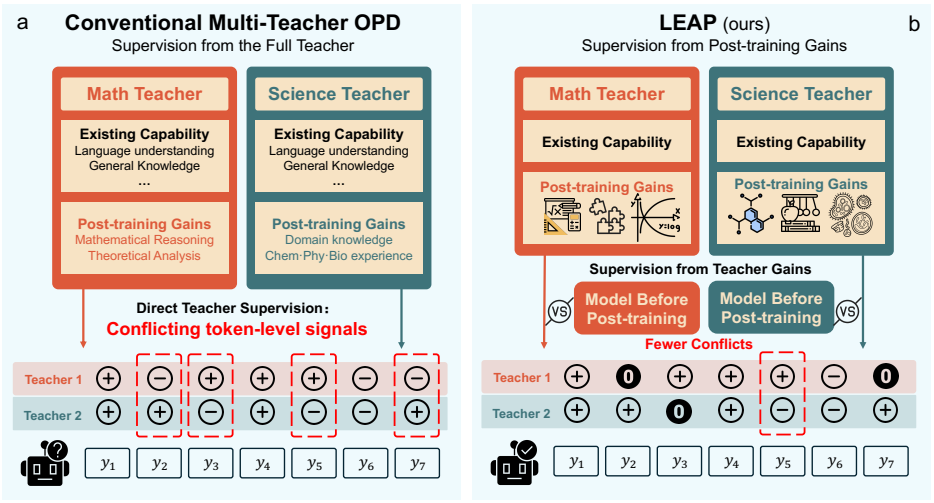

Figure 1: Overview of LEAP: Learning from Experts' Acquired Proficiency.

This distinction is particularly important in cross-domain distillation. A mathematics expert and a science expert need not agree on their complete next-token distributions, which contain substantial shared linguistic, formatting, and pretrained knowledge. Capability fusion instead requires the *specialization-induced changes* of these experts to be transferred without unnecessarily interfering with one another. LEAP operationalizes this principle directly on student-generated trajectories.

Our experiments support this view at both the token and model levels. On identical student rollouts, LEAP reduces opposite-sign supervision between two expert teachers from 4.4% under standard OPD to 1.6%, with most raw conflicts either reconciled by base centering or filtered by the consistency gate. In controlled sequential-distillation experiments, LEAP outperforms standard OPD across all evaluated benchmarks. Moreover, under increasingly imbalanced math+science and math+coding training mixtures, standard OPD exhibits substantial cross-domain degradation, whereas LEAP remains markedly more robust.

Our main contributions are:

- We identify **spurious teacher conflict** as an important source of interference in multi-teacher OPD, and show that raw teacher–student signals can obscure the specialization-specific knowledge contributed by different experts.

- We propose **LEAP**, a multi-teacher on-policy delta distillation framework that isolates expert-specific capability increments through teacher–base centering and consistency-based filtering.

- Through token-level mechanistic analysis and controlled multi-domain distillation experiments, we show that LEAP substantially improves signal compatibility and reduces cross-domain interference compared with standard OPD.

2 Related Work

2.1 On-Policy Distillation

On-policy distillation (OPD) mitigates distribution mismatch by supervising students on their own generated trajectories, as developed in GKD and MiniLLM (Agarwal et al., 2024; Gu et al., 2023). Recent work extends this paradigm through different sources and formulations of supervision. On-Policy Self-Distillation (OPSD) constructs a teacher by conditioning the model on privileged information, such as reference solutions, and transfers this guidance to the student without access to that information (Zhao et al., 2026). Generative Adversarial Distillation (GAD) enables black-box OPD through a learned discriminator, supporting transfer across tokenizers without access to teacher logits (Ye et al., 2025).

Another line of work examines what makes on-policy supervision effective. ExOPD introduces reward extrapolation and flexible reference policies within a KL-regularized RL framework (Yang et al., 2026), while Uni-OPD combines improved student exploration with outcome-guided calibration of teacher supervision (Hou et al., 2026). Rethinking OPD identifies compatible thinking patterns and novel teacher capabilities as key conditions for successful transfer (Li et al., 2026). Beyond matching a teacher's final policy, OPD^2 and Direct-OPD transfer policy changes induced by post-training (Heo et al., 2026; Feng et al., 2026). LEAP builds on this perspective but focuses on supervision conflict during multi-expert consolidation: how signals from different specialized teachers interact when their capabilities are transferred into a shared student. Specifically, we investigate how the shared base–student discrepancy can induce opposite-signed teacher–student signals even when the teachers' specialization deltas agree, and examine the implications of base-referenced supervision for capability retention and acquisition under sequential and domain-routed mixed training. Our emphasis is thus on characterizing and mitigating interference during expert consolidation, extending the study of policy-shift transfer to the compatibility of supervision across experts.

2.2 Multi-Teacher Distillation

Multi-teacher distillation consolidates specialized capabilities into a single student. MiMo-V2-Flash introduces Multi-Teacher On-Policy Distillation (MOPD) as a post-training paradigm that integrates independently trained domain experts through dense supervision on student-generated trajectories (Xiao et al., 2026). DeepSeek-V4 and Nemotron 3 Ultra further scale this paradigm to more than ten specialized teachers, demonstrating its applicability to large-scale capability integration (Xu et al., 2026; Blakeman et al., 2026). ExOPD also investigates integration across independently trained domain experts (Yang et al., 2026), while Direct-OPD demonstrates sequential composition of teacher policy shifts (Feng et al., 2026).

These advances motivate a closer examination of how expert supervision signals interact. LEAP studies token-level interference by evaluating experts on the same student-generated trajectories and separating specialization-induced signals from the shared base–student discrepancy. Our analysis attributes a substantial portion of apparent cross-teacher conflict to this shared component, motivating the joint composition of base-referenced specialization updates rather than full teacher–student corrections.

3 METHODS

3.1 ALGORITHM

We consider on-policy distillation where the student generates trajectories and receives dense token-level supervision from one or more expert teachers. Let $s_t = (x, y_{<t})$ denote the state at step t, and let y_t be the token sampled from the student policy π_θ.

Standard On-Policy Distillation For a teacher T, standard OPD assigns each sampled token the log-probability difference between the teacher and the student:

$$r_t^{\mathrm{OPD}} = \log p_T(y_t \mid s_t) - \log \pi_\theta(y_t \mid s_t).$$

This signal directly measures how much more or less the teacher favors the student's sampled action. In the multi-teacher setting, however, each teacher independently compares its full predictive distribution against the student, which can introduce conflicting supervision.

Base-Referenced Delta Signal LEAP instead transfers the policy change acquired during expert post-training. Let B_T denote the pre-RL base model corresponding to teacher T. We first compute

$$\delta_T(s_t, a) = \log p_T(a \mid s_t) - \log p_{B_T}(a \mid s_t), \qquad \delta_t = \delta_T(s_t, y_t).$$

We construct the supervision signal using the action-value–state-value decomposition of an advantage. Specifically, we treat $\delta_T(s, a)$ as a local action-value surrogate, $\widetilde{Q}_T(s, a) = \delta_T(s, a)$, and define its state baseline under the student's current policy as

$$\widetilde{V}_{T,\theta}(s) = \mathbb{E}_{a \sim \pi_\theta(\cdot|s)}[\widetilde{Q}_T(s, a)].$$

The resulting advantage surrogate is

$$r_t^{\Delta} = \widetilde{Q}_T(s_t, y_t) - \widetilde{V}_{T,\theta}(s_t).$$

Thus, the signal measures whether the sampled token receives a larger specialization-induced increase than the student's average action at the same state. This provides an advantage-based interpretation of the centering operation also used by Heo et al. (2026).

In practice, the expectation is approximated over the top-K student tokens with $K = 128$, renormalized according to the student probabilities. We additionally apply a direction-consistency gate. Define the centered teacher–student direction

$$d_t = [\log p_T(y_t \mid s_t) - \log \pi_\theta(y_t \mid s_t)] - \mathbb{E}_{a \sim \pi_\theta(\cdot|s_t)} [\log p_T(a \mid s_t) - \log \pi_\theta(a \mid s_t)] .$$

The final LEAP signal is

$$\hat{r}_t^{\mathrm{LEAP}} = \begin{cases} r_t^{\Delta}, & r_t^{\Delta} d_t \geqslant 0, \\ 0, & r_t^{\Delta} d_t < 0. \end{cases}$$

3.2 TRAINING SETUP

Models We use **Qwen3-1.7B** as the student model. Our expert teachers are **M2RL-RL_Math** and **M2RL-RL_Science**, both obtained by reinforcement-learning post-training from **Qwen3-4B-Base** (Wang et al., 2026). We therefore use Qwen3-4B-Base as the shared teacher base B_T for both experts.

The shared initialization of the two teachers is important for LEAP: their policy shifts are measured relative to the same reference model, making the resulting specialization deltas directly comparable. All models belong to the Qwen3 family and share the same vocabulary, avoiding cross-tokenizer alignment issues.

3.3 TRAINING DATA

We use the math and science subsets of `nvidia_open_reasoning_balanced_100k` (Moshkov et al., 2025; Ahmad et al., 2025; NVIDIA, 2025), containing 33,333 examples for each

domain. Since the two subsets originate from the same data source and preprocessing pipeline, they provide a controlled setting for studying cross-domain transfer.

For mixed-domain experiments, we construct 16K-example datasets with math-to-science ratios:

$$25{:}2, \quad 1{:}1, \quad 2{:}25.$$

Each example retains its domain label for teacher routing.

Optimization Both OPD and LEAP use the same training configuration. We optimize the student using the GRPO PPO-clipped objective, with the dense token-level advantage defined as

$$A_t = 0.2\, r_t,$$

where r_t is either the OPD or LEAP signal.

We use a batch size of 256 trajectories per step, a learning rate of 5×10^{-6}, and train for 50 optimization steps per stage. The maximum completion length is 8192.

We additionally use three stabilization choices consistently across all methods: a KL coefficient of $\beta = 0.05$, teacher-side prompt alignment with empty thinking blocks removed, and EOS-position loss masking.

3.4 Mixed-Domain Multi-Teacher Training

We further study the more challenging setting in which examples from two domains coexist within the same training batch. We consider two teacher configurations: Math + Science and Math + Coding. For each configuration, let $d \in \{\text{Science}, \text{Coding}\}$ denote the domain paired with mathematics. Each training example is routed to its corresponding domain expert according to its dataset label:

$$T(x) = \begin{cases} T_{\text{Math}}, & x \in \mathcal{D}_{\text{Math}}, \\ T_d, & x \in \mathcal{D}_d. \end{cases}$$

Within each configuration, both teachers share the same base model, Qwen3-4B-Base, and LEAP computes the specialization delta of the routed teacher relative to this shared reference. At each optimization step, mathematics examples and examples from domain d are jointly sampled into the same batch, and their teacher-guided objectives contribute to a single student update. We vary the mathematics-to-d sampling ratio across 25:2, 1:1, and 2:25, and evaluate the student after 50 training steps. These configurations allow us to assess whether base-referenced specialization signals remain composable across different domain pairings and mixture ratios in the presence of potential cross-domain gradient interference.

4 Experiment

In this section, we start from a math teacher and a science teacher, which is common during multi-teacher on policy distillation and go further into a math teacher and a science/coding teacher. We first analyze token-level supervision conflicts between math and science teachers in Section 4.1. We then evaluate capability acquisition and retention under sequential distillation in Section 4.2. Finally, Section 4.3 examines mixed-domain distillation across different sampling ratios for Math + Science and Math + Coding.

We evaluate mathematical reasoning ability on AIME 2024 and AIME 2025 (Jia, 2024; math ai, 2025), scientific reasoning ability on GPQA-Diamond (Rein et al., 2023) and SciBench (Wang et al., 2023), and coding ability on LiveCodeBench v5 (LCBv5) (Jain et al., 2025).

4.1 Analyzing Conflicts in Multi-Teacher Distillation

We first investigate whether transferring teachers' post-training changes can reduce conflicting supervision in multi-teacher distillation. We collect student rollouts using Qwen3-1.7B on math and science prompts, yielding 97,884 completion tokens, and evaluate both domain teachers on identical

trajectories. Among tokens where both raw OPD signals are active ($|s| > 10^{-4}$), 4.44% receive opposite-signed signals. To understand these conflicts, we decompose the raw signal from teacher T as

$$\underbrace{\log p_T - \log p_S}_{\text{raw OPD}} = \underbrace{\log p_T - \log p_B}_{\text{post-training delta}} + \underbrace{\log p_B - \log p_S}_{\text{shared offset}}, \tag{3}$$

where B is the teachers' shared base model, S is the student, and all probabilities refer to the same token and context. The shared offset can move two same-signed post-training deltas to opposite sides of zero. Thus, conflicting raw signals do not necessarily imply disagreement between the teachers' post-training changes.

Among the 1,885 tokens with conflicting raw signals, simply subtracting the teacher-base log probability restores sign agreement for **84.2%**. Inspired by Heo et al. (2026), adding the student-expectation correction increases this proportion to **87.9%** (+3.7 percentage points). Directional gating subsequently suppresses selected signals, leaving **7.5%** of the original conflict tokens still conflicting. Across all tokens, approximately 75% of each teacher's signals are zeroed by gating. These results distinguish sign alignment through delta extraction from conflict removal through abstention.

The complete signal exhibits lower conflict rates at three granularities (Table 1). Overall, the token conflict rate decreases from 4.44% to 1.58%. Across 155 trajectories containing at least 50 tokens, 83.9% show a reduction, and the median trajectory-level rate decreases from 4.0% to 1.3%. Reductions also hold separately for each domain's trajectories.

Table 1: Sign-conflict rates on identical student rollouts.

Subset	Raw OPD	Post-training delta
All	4.44%	**1.58%**
Math	5.12%	**1.47%**
Science	3.05%	**1.86%**

Conflicts occur across token categories, including words, numbers, punctuation, and markup. Strong conflicts are rare: only 25 of 13,362 tokens with both raw signal magnitudes above 0.5 have opposite signs. Our findings therefore support a mechanism for reducing conflicting supervision, rather than suggesting that strong conflicts are pervasive.

Case study: a shared offset induces conflicting signals. Table 2 illustrates this mechanism using a sentence-initial token, `The`. Relative to their shared base model, both teachers increase the token's log probability, yielding positive post-training deltas of $+0.66$ and $+1.71$. However, the shared base–student offset is -1.64. Its magnitude lies between the two deltas, shifting the math teacher's raw signal below zero while leaving the science teacher's signal positive. Consequently, the raw OPD signals disagree even though the post-training changes agree in direction. Subtracting the shared offset recovers this agreement.

Table 3 presents representative conflicting signals on tokens from different rollouts. Examples include sentence openings, connectives, and derivation transitions. While raw OPD signals disagree in sign, our method either produces same-signed signals or suppresses one side through directional gating.

Table 2: Signal decomposition for a sentence-initial `The` token. Both teachers have positive post-training deltas, but the shared base–student offset induces opposite-signed raw OPD signals. Values are rounded.

Component	Math teacher	Science teacher
LEAP: $\log p_T - \log p_B$	$+0.66$	$+1.71$
Shared offset: $\log p_B - \log p_S$	-1.64	-1.64
Raw OPD: $\log p_T - \log p_S$	-0.99	$+0.07$

Reduced sign conflict suggests a potential mechanism for mitigating cross-domain interference. When different teachers supervise the same state–action pair, same-signed signals encourage compatible local updates to the student. If nearby states also induce similar student policy gradients,

Table 3: Representative conflicting tokens with patterns from distinct rollouts. † denotes a gated signal; leading token spaces are omitted.

		Raw OPD		LEAP	
Context	Token	Math	Science	Math	Science
Sentence start	`The`	-1.44	$+0.99$	$+8.65$	$+7.92$
Derivation transition	`Compute`	-0.57	$+0.87$	$+5.37$	$+6.27$
Conclusion connective	`Thus`	-0.54	$+1.13$	$+4.92$	$+4.70$
Sentence start	`We`	$+0.60$	-0.52	$+3.75$	$0^{\dagger}$
Pattern description	`pattern`	-0.56	$+0.91$	$+1.98$	$+3.21$
Re-derivation	`compute`	-0.56	$+1.07$	$+1.37$	$+4.40$
Question reformulation	`regarding`	-1.82	$+0.60$	$0^{\dagger}$	$+2.64$
Counting step	`count`	-1.00	$+0.52$	-1.25	$0^{\dagger}$

improved sign agreement can make updates from different teachers less likely to oppose one another, providing an intuitive basis for more compatible learning across domains. Token-level sign agreement alone, however, does not guarantee reduced interference between updates on different trajectories. We therefore examine the behavioral implications of this mechanism empirically in the following two sections, through sequential expert distillation and mixed-domain training with varying domain ratios.

4.2 Sequential Distillation across Domains

Building on the signal-conflict analysis, we examine whether transferring post-training increments helps students acquire capabilities from a subsequent expert while retaining those learned previously. We first distill from the math teacher for 50 steps, then continue from each method's step-50 checkpoint with the science teacher for another 50 steps. This teacher switch allows us to assess both the retention of acquired mathematical capabilities and the acquisition of scientific capabilities. LEAP and standard OPD use the same student initialization (Qwen3-1.7B), teachers, training data, and hyperparameters, differing only in the distillation signal.

Table 4: Benchmark accuracy (%) before training, after 50 steps of math distillation, and after an additional 50 steps of science distillation.

		Math phase		Science phase	
	Baseline	Step 50		Step 100	
Benchmark	Step 0	OPD	LEAP	OPD	LEAP
AIME24	16.7	33.3	**50.0**	30.0	**53.3**
AIME25	6.7	23.3	**36.7**	20.0	**43.3**
GPQA-Diamond	28.8	31.8	**38.9**	39.4	**40.4**
SciBench	26.9	35.2	**42.6**	36.0	**43.3**

Figure 2 and Table 4 show that LEAP achieves higher math accuracy during the first phase. Beyond this initial advantage, the two methods exhibit different retention behavior after switching to the science teacher. From step 50 to step 100, OPD's AIME24 and AIME25 scores each decrease by 3.3 percentage points. In contrast, LEAP preserves its previously acquired math performance, with scores increasing from 50.0% to 53.3% on AIME24 and from 36.7% to 43.3% on AIME25.

This retention is accompanied by continued improvement on the science benchmarks. During the second phase, LEAP's GPQA-Diamond accuracy increases from 38.9% to 40.4%, and its SciBench accuracy increases from 42.6% to 43.3%. OPD also improves on both science benchmarks, finishing at 39.4% and 36.0%, respectively, but with a small decline in math performance. The SciBench trajectories further show that OPD's early improvement after the teacher switch partially recedes, whereas LEAP maintains its gains more consistently.

Together, these results suggest that, in this sequential multi-teacher setting, transferring post-training increments supports the incorporation of a subsequent expert's capabilities while preserving those acquired earlier.

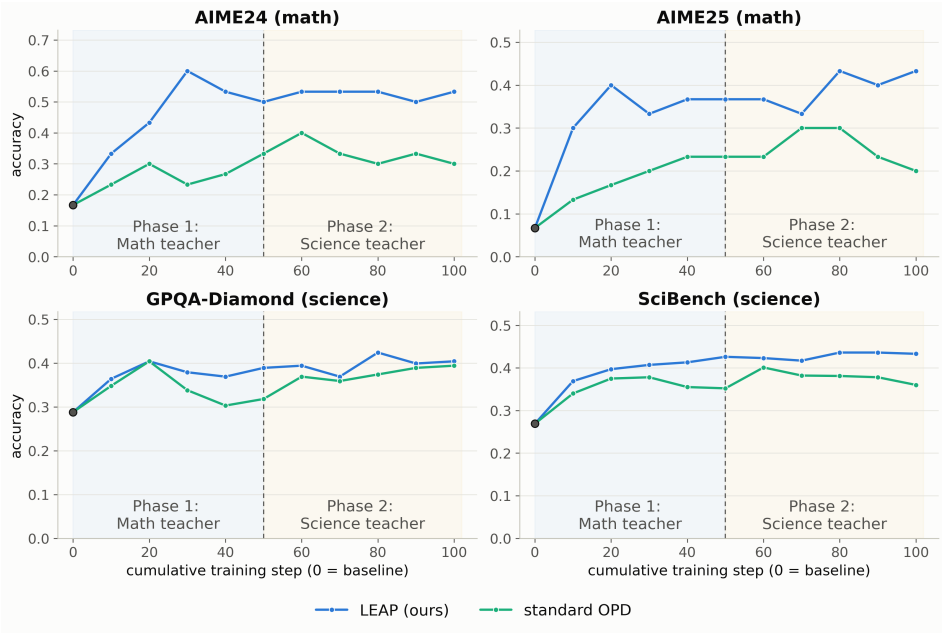

Figure 2: Sequential distillation with LEAP (ours) and standard OPD across four benchmarks.

4.3 Distillation with Mixed Domain Ratios

We next examine whether LEAP's benefits extend to mixed-domain distillation across different domain pairings. We consider two configurations: Math + Science and Math + Coding. In each configuration, examples from both domains coexist within the same training batch and each of them is routed to its corresponding domain teacher. Both teachers are derived from Qwen3-4B-Base, which serves as the shared reference for computing LEAP's teacher specialization deltas. We train the student for 50 steps using mathematics-to-science or mathematics-to-coding sampling ratios of 25:2, 1:1, and 2:25.

Figure 3 compares accuracy at step 50 against standard OPD and the untrained student baseline, and Figure 4 shows the corresponding learning trajectories. Across both configurations, LEAP outperforms OPD on every evaluated benchmark at all three ratios in the reported main runs. For Math + Science, the improvements range from 10.0 to 23.3 percentage points on the mathematics benchmarks and from 3.0 to 6.2 points on the science benchmarks. Even under the science-heavy 2:25 mixture, LEAP achieves 30.0% on AIME24 and 26.7% on AIME25, compared with 16.7% and 10.0% for OPD, while also improving GPQA-Diamond and SciBench by 5.1 and 3.0 points, respectively. Detailed numerical results for both domain pairings and checkpoint-level results are provided in Appendix A.

The Math + Coding configuration shows a similar advantage. LEAP improves over OPD by 6.6–30.0 percentage points on the mathematics benchmarks and by 6.8–8.4 points on LCBv5. Under the math-heavy 25:2 mixture, LEAP achieves 53.3% on AIME24 and 40.0% on AIME25, compared with 23.3% on both benchmarks for OPD, while also improving LCBv5 from 5.2% to 13.6%. Under the coding-heavy 2:25 mixture, LEAP reaches 15.2% on LCBv5 versus 7.9% for OPD and retains an advantage on both mathematics benchmarks. Together, these results support the effectiveness of base-referenced specialization signals across different domain pairings and substantially different mixture ratios.

5 Conclusion

We presented LEAP, a multi-teacher on-policy distillation framework that transfers experts' post-training increments. Our analysis shows that opposite-signed teacher–student signals can arise even

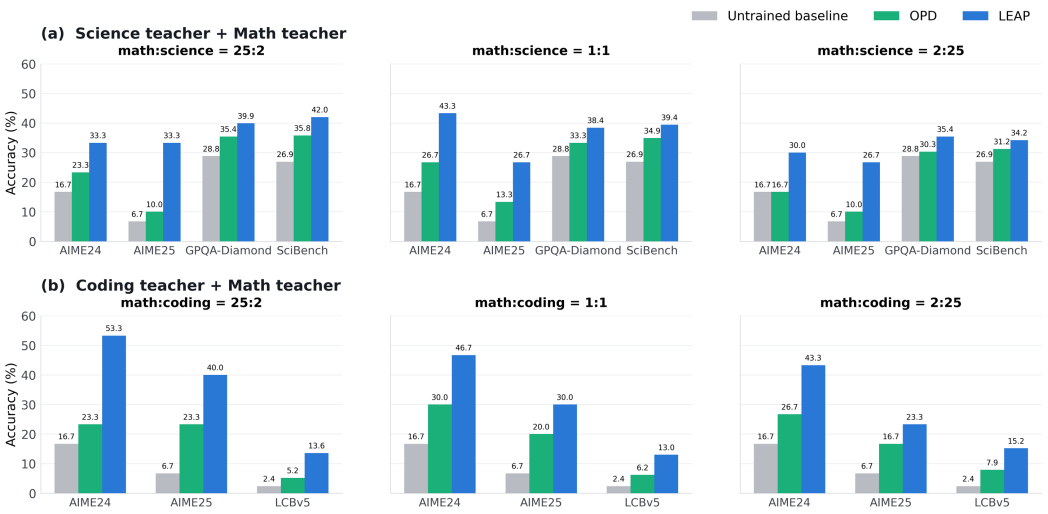

Figure 3: Final benchmark accuracy (%) after 50 training steps under mixed-domain distillation. The top group shows Math + Science, and the bottom group shows Math + Coding. Within each group, the three panels correspond to mathematics-to-science or mathematics-to-coding ratios. Gray, green, and blue bars denote the untrained student baseline, standard OPD, and LEAP.

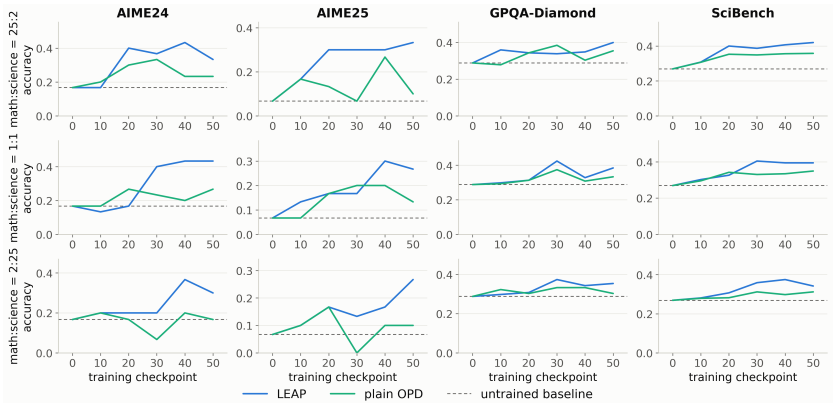

Figure 4: Learning trajectories for Math + Science distillation. Rows correspond to math-to-science ratios, columns correspond to four benchmarks' accuracy.

when experts' post-training changes agree in direction. Empirically, LEAP reduces token-level supervision conflicts and outperforms standard OPD across the evaluated domain pairings and training mixtures. In sequential distillation, it preserves previously acquired mathematical capabilities while continuing to improve on science benchmarks. In domain-routed mixed training, it achieves higher performance across balanced and imbalanced query mixtures. These findings support post-training increments as useful supervision for multi-expert capability integration. Future work can examine whether these benefits extend to larger collections of experts, heterogeneous model families, and longer training horizons.

Disclosure of AI Use

We used a large language model solely to polish the language and improve the readability of the manuscript. The authors reviewed and edited all AI-assisted revisions and take full responsibility for the final content.

Reproducibility statement

The anonymous source code for reproducing our experiments is available at: https://anonymous.4open.science/r/LEAP_iclr2027/

References

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 *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Wasi Uddin Ahmad, Sean Narenthiran, Somshubra Majumdar, Aleksander Ficek, Siddhartha Jain, Jocelyn Huang, Vahid Noroozi, and Boris Ginsburg. Opencodereasoning: Advancing data distillation for competitive coding. 2025. URL https://arxiv.org/abs/2504.01943.

Aaron Blakeman, Aaron Thomas, Aastha Jhunjhunwala, Abhibha Gupta, Abhinav Khattar, Adam Rajfer, Adi Renduchintala, Adil Asif, Aditya Vavre, Adriana Flores Miranda, et al. Nemotron 3 ultra: Open, efficient mixture-of-experts hybrid mamba-transformer model for agentic reasoning. *arXiv preprint arXiv:2606.15007*, 2026.

Shiyuan Feng, Huan-ang Gao, Haohan Chi, Hanlin Wu, Zhilong Zhang, Zheng Jiang, Bingxiang He, Wei-Ying Ma, Ya-Qin Zhang, and Hao Zhou. Weak-to-strong generalization via direct on-policy distillation. *arXiv preprint arXiv:2607.05394*, 2026.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: On-policy distillation of large language models. *arXiv preprint arXiv:2306.08543*, 2023.

Byeongho Heo, Jaehui Hwang, Sangdoo Yun, and Dongyoon Han. On-policy delta distillation. *arXiv preprint arXiv:2607.15161*, 2026.

Wenjin Hou, Shangpin Peng, Weinong Wang, Zheng Ruan, Yue Zhang, Zhenglin Zhou, Mingqi Gao, Yifei Chen, Kaiqi Wang, Hongming Yang, et al. Uni-opd: Unifying on-policy distillation with a dual-perspective recipe. *arXiv preprint arXiv:2605.03677*, 2026.

Naman Jain, 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 *International Conference on Learning Representations*, volume 2025, pp. 58791–58831, 2025.

Maxwell Jia. Aime problem set 2024, 2024. URL https://huggingface.co/datasets/Maxwell-Jia/AIME_2024.

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

math ai. Aime problem set 2025, 2025. URL https://huggingface.co/datasets/math-ai/aime25.

Ivan Moshkov, Darragh Hanley, Ivan Sorokin, Shubham Toshniwal, Christof Henkel, Benedikt Schifferer, Wei Du, and Igor Gitman. Aimo-2 winning solution: Building state-of-the-art mathematical reasoning models with openmathreasoning dataset. *arXiv preprint arXiv:2504.16891*, 2025.

NVIDIA. Opensciencereasoning-2. https://huggingface.co/datasets/nvidia/OpenScienceReasoning-2, 2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

Haoqing Wang, Xiang Long, Ziheng Li, Yilong Xu, Tingguang Li, and Yehui Tang. To mix or to merge: Toward multi-domain reinforcement learning for large language models. In *Third Conference on Language Modeling*, 2026. URL https://openreview.net/forum?id=jP7j5XkG8J.

Xiaoxuan Wang, Ziniu Hu, Pan Lu, Yanqiao Zhu, Jieyu Zhang, Satyen Subramaniam, Arjun R Loomba, Shichang Zhang, Yizhou Sun, and Wei Wang. Scibench: Evaluating college-level scientific problem-solving abilities of large language models. *arXiv preprint arXiv:2307.10635*, 2023.

Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, et al. Mimo-v2-flash technical report. *arXiv preprint arXiv:2601.02780*, 2026.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

Wenkai Yang, Weijie Liu, Ruobing Xie, Kai Yang, Saiyong Yang, and Yankai Lin. Learning beyond teacher: Generalized on-policy distillation with reward extrapolation. *arXiv preprint arXiv:2602.12125*, 2026.

Tianzhu Ye, Li Dong, Zewen Chi, Xun Wu, Shaohan Huang, and Furu Wei. Black-box on-policy distillation of large language models. *arXiv preprint arXiv:2511.10643*, 2025.

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

A Additional Results for Mixed-Domain Distillation

A.1 Checkpoint-Level Results

Table 5 reports checkpoint-level accuracy for Math + Science distillation under three sampling ratios. Each paired entry lists LEAP followed by OPD. These results provide the numerical values underlying Figure 4.

Table 5: Checkpoint-level accuracy for Math + Science distillation. Each paired entry reports LEAP / OPD as accuracy fractions. The baseline is the untrained Qwen3-1.7B student.

Math:Science	Step	AIME24	AIME25	GPQA-Diamond	SciBench
Baseline		0.167	0.067	0.288	0.269
25:2	10	0.167/0.200	0.167/0.167	0.359/0.278	0.308/0.307
	20	0.400/0.300	0.300/0.133	0.343/0.343	0.400/0.353
	30	0.367/0.333	0.300/0.067	0.338/0.384	0.387/0.349
	40	0.433/0.233	0.300/0.267	0.348/0.303	0.407/0.356
	50	0.333/0.233	0.333/0.100	0.399/0.354	0.420/0.358
1:1	10	0.133/0.167	0.133/0.067	0.298/0.293	0.302/0.295
	20	0.167/0.267	0.167/0.167	0.313/0.313	0.326/0.342
	30	0.400/0.233	0.167/0.200	0.424/0.374	0.404/0.330
	40	0.433/0.200	0.300/0.200	0.328/0.308	0.394/0.334
	50	0.433/0.267	0.267/0.133	0.384/0.333	0.394/0.349
2:25	10	0.200/0.200	0.100/0.100	0.298/0.323	0.281/0.279
	20	0.200/0.167	0.167/0.167	0.308/0.303	0.307/0.282
	30	0.200/0.067	0.133/0.000	0.374/0.333	0.359/0.312
	40	0.367/0.200	0.167/0.100	0.343/0.333	0.375/0.298
	50	0.300/0.167	0.267/0.100	0.354/0.303	0.342/0.312

A.2 Final Results Across Domain Pairings

Tables 6 and 7 report accuracy after 50 training steps for Math + Science and Math + Coding, respectively. Both tables compare LEAP with OPD under three sampling ratios and include the untrained student as a reference.

Table 6: Final accuracy (%) for Math + Science distillation. Ratios denote math:science. Bold indicates the higher accuracy between OPD and LEAP within each ratio.

Ratio	Method	AIME24	AIME25	GPQA-Diamond	SciBench
—	Untrained	16.7	6.7	28.8	26.9
25:2	OPD	23.3	10.0	35.4	35.8
	LEAP	**33.3**	**33.3**	**39.9**	**42.0**
1:1	OPD	26.7	13.3	33.3	34.9
	LEAP	**43.3**	**26.7**	**38.4**	**39.4**
2:25	OPD	16.7	10.0	30.3	31.2
	LEAP	**30.0**	**26.7**	**35.4**	**34.2**

Table 7: Final accuracy (%) for Math + Coding distillation. Ratios denote math:coding. Bold indicates the higher accuracy between OPD and LEAP within each ratio.

Ratio	Method	AIME24	AIME25	LCBv5
—	Untrained	16.7	6.7	2.4
25:2	OPD	23.3	23.3	5.2
	LEAP	**53.3**	**40.0**	**13.6**
1:1	OPD	30.0	20.0	6.2
	LEAP	**46.7**	**30.0**	**13.0**
2:25	OPD	26.7	16.7	7.9
	LEAP	**43.3**	**23.3**	**15.2**