PADD: Path-Aligned Decompression Distillation for Non-Router Teacher to Guide MoE Student Learning

Xinyue Peng[1] Yi Qian[1] Jiaojiao Lin[1] Wenjian Shao[1] Yanming Liu[2]
[1]Intel Corporation, China [2]Zhejiang University, China

Keywords: Mixture-of-Experts • Knowledge Distillation • Dense-to-MoE • GRPO • Mathematical Reasoning

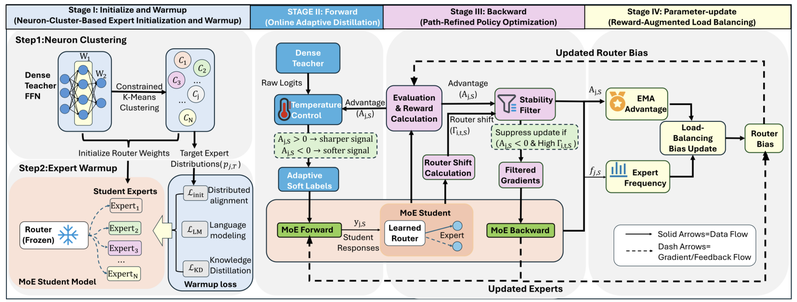
Abstract

Mixture-of-Experts (MoE) language models scale capacity without scaling compute, but they need a **router** to assign tokens to experts — and a dense reasoning teacher *has no router* to teach with. We ask: *can a small dense teacher distill its 7B-grade reasoning signal into a sparsely activated MoE student, and have the student* ***surpass the teacher at the same inference cost****?*

PADD (**P**ath-**A**ligned **D**ecompression **D**istillation) answers yes. The teacher FFN is treated as a latent mixture — **Stage I** runs cardinality-constrained K-Means on its neurons to seed the student experts, and a frozen uniform router warms them up. **Stages II–IV** then run jointly on a single training loop: adaptive KD, path-regularized GRPO, and reward-augmented load-balance.

Result. On a Qwen-3B active MoE student with a Qwen2.5-Math-7B teacher, PADD reaches **80.2%** average Pass@1 — **+2.5 pt** over the 7B teacher at **0.47×** the per-token FLOPs, while keeping broad-skill accuracy within 0.2 pt of the untrained base.

Method



PADD framework. Phase A: cluster teacher FFN → student experts. Phase B: one-loop training (II → III → IV).

Phase A (offline, once). Cluster teacher FFN neurons by co-activation. Each cluster becomes one student expert. Warm up with a frozen **uniform router** (one epoch) so the student starts with differentiated experts, not random cold-start noise.

Phase B (online, every step). One training loop on the math corpus $\mathcal{D}_C$:

II **Adaptive KD.** Re-temper teacher logits by the student's *path advantage* $A = (r - \bar{r})/\sigma_r$: the temperature $\tau \cdot \Phi(A)$ is loud on paths where the student is currently strong, and quiet on weak paths.

III **PR-GRPO.** Reweight the policy ratio by the routing shift Γ on negative-advantage steps. The factor $\exp(-\lambda\Gamma)$ suppresses updates that destabilise the router.

IV **Reward-augmented load balance.** EMA-smoothed per-expert reward enters the router bias via $b \leftarrow b + \eta(f - \bar{f}) + \gamma\,\mathrm{EMA}(A)$, so useful experts stay warm.

Why does Stage I matter? The teacher FFN encodes a latent module structure — co-activating neurons belong to the same functional module. Cardinality-constrained K-Means partitions them into equal-size clusters; each seeds one student expert. After Stage I, **NMI** between teacher clusters and student expert activations rises to **0.030** (vanilla 0.013, $p < 0.01$); **ESI** 0.029 vs 0.014. The student starts with differentiated experts aligned to the teacher's functional decomposition — rather than re-discovering expert roles online, which is exactly what wastes RL samples in prior upcycling-based methods.

Why PADD wins

vs. prior work (one line each).

- **vs. GSPO / RSPO / Online KD:** PADD starts from a router-aware expert decomposition (Stage I) — the others start from random upcycling and must re-discover expert roles online, so they waste RL samples on routing churn.
- **vs. Vanilla-GRPO:** PR-GRPO's path-aware reweighting suppresses the routing-shift spikes that destroy specialization (47% lower Γ at step 600).
- **vs. Dense teacher:** the student's per-token FLOPs are **0.47×** the teacher's, yet Pass@1 is **+2.5 pt** higher on Qwen avg — capacity without cost.

The four stages, in one line.

I *Seed:* K-Means teacher FFN → student experts.
II *Adaptive KD:* $\tau \cdot \Phi(A)$ re-temper on path advantage.
III *PR-GRPO:* $\exp(-\lambda\Gamma)$ reweight on bad-path steps.
IV *Load balance:* $b \leftarrow b + \eta(f - \bar{f}) + \gamma\,\mathrm{EMA}(A)$.

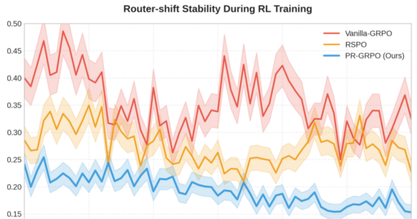
Experiments

Setup. Two students (Qwen-3B / 30.5B MoE; DeepSeek-V2-Lite 2.4B / 15.7B), each distilled from a same-family 7B dense math teacher. Compared against *Base* (upcycled), and three published distillation/RL baselines (*GSPO*, *RSPO*, *Online KD*) on five math benchmarks: AIME24, AMC, MATH-500, Minerva, OlympiadBench. All baselines use the same compute budget and are trained from the same MoE initialisation.

Headline. PADD matches or exceeds the 7B dense teacher on **4/5** Qwen and **2/5** DeepSeek benchmarks. On OlympiadBench, PADD scores **70.7%** — **+6.9 pt** over Base, **+12.5 pt** over the strongest baseline (RSPO 66.2%).

Ablation. Stage I is the single largest contributor (−10.4 pt on OlympiadBench without it); Stages II–IV each contribute 1.5–4 pt. Removing all of II–IV together is roughly equivalent to removing Stage I alone — highlighting that expert decomposition is the dominant factor, but the online-stability machinery matters for the last few points.

Why PR-GRPO? Router-shift Γ (per-step L2 change in router weights) stays low and tight under PR-GRPO: **0.18** at step 600 vs 0.34 for Vanilla-GRPO (47% lower). The ECDF right tail confirms fewer extreme shifts — exactly the updates that break specialization. PR-GRPO's path-aware reweighting suppresses routing-shift spikes during policy updates, so the router converges to stable assignments instead of oscillating across training rounds.



Router-shift Γ over 600 RL steps.

Generalization. Math-only training does *not* erode broad skills: Qwen non-math avg **52.0** vs untrained Base 52.2 (−0.2 pt); DeepSeek non-math 38.5 vs 38.9 (−0.4 pt). Vanilla-GRPO drops to 49.3 / 37.1 — showing that PADD's stable routing also protects non-math capabilities.

Compute. PADD costs about 24 GPU-hours on 8×H100 for the Qwen student, of which 79% is the unified Stage II–IV forward/backward. Stage I K-Means is offline (one-shot, <10 min, single A100). Per-token inference FLOPs drop to **0.47×** the 7B dense teacher — the same activated-parameter budget, with sparse routing cutting the rest.

Reproducibility. PADD recipe, configs, and trained student checkpoints will be released; training log and ablation spreadsheets are in the paper supplementary.

Results

Table 1. Main math results (Pass@1 %). Best per column in **bold**.

Qwen (3.3B active)						
Method	AIME	AMC	M500	Min.	Oly.	Avg
Teacher	83.0	94.7	91.3	55.5	63.8	77.7
Base	74.5	89.6	90.5	47.5	62.2	72.9
GSPO	80.4	94.8	93.7	49.1	63.6	76.3
RSPO	80.3	95.2	94.1	50.4	66.2	77.2
Online KD	78.4	86.1	92.1	51.6	59.6	73.6
PADD	**83.0**	**95.9**	**96.4**	**55.0**	**70.7**	**80.2**

DeepSeek (2.4B active)						
Method	AIME	AMC	M500	Min.	Oly.	Avg
Teacher	55.3	69.2	79.5	36.8	49.7	58.1
Base	38.4	49.6	37.8	28.3	31.7	37.2
GSPO	54.8	69.3	57.6	36.2	47.9	53.2
RSPO	55.7	69.8	58.4	37.9	49.6	54.3
Online KD	50.1	59.5	47.8	32.3	43.7	46.7
PADD	**57.6**	**69.5**	**59.3**	**39.8**	**49.7**	**55.2**

Discussion of results

What the table actually says. On Qwen, PADD matches the 7B dense teacher on AIME (83.0 vs 83.0) and exceeds it on AMC (95.9 vs 94.7), MATH-500 (96.4 vs 91.3) and OlympiadBench (70.7 vs 63.8) — the larger the benchmark, the bigger the margin. The smallest gap is on Minerva (55.0 vs 55.5), where wordy STEM reasoning puts more weight on language modelling than on multi-step chaining — exactly the regime where MoE sparsity helps least.

Why Qwen > DeepSeek. The DeepSeek-V2-Lite teacher is itself weaker (58.1 vs 77.7 average), and its expert layout differs more from the student's initialisation — so Stage I has less teacher signal to extract and the online loop has more to do. PADD still matches the teacher on 2/5 benchmarks and beats every baseline, but the absolute headroom over the teacher is smaller.

Ablation reading. Removing Stage I costs 10.4 pt on OlympiadBench — the same as removing all of II–IV together. Interpretation: getting the expert decomposition right is the dominant factor, and the online-stability machinery (II–IV) is what turns that good decomposition into the last few points. Skip Stage I, you fight routing noise; skip II–IV, you plateau early.

Stability curve. PR-GRPO's Γ curve (blue) is lower *and* tighter than Vanilla-GRPO across all 600 steps — 47% lower mean, and the right tail is visibly shorter (the ECDF inset). These spikes are exactly where Vanilla-GRPO's expert identities drift; suppressing them preserves the specialisation that Stage I built.

Compute profile. Stage I is a one-shot offline pass (<10 min, single A100); Stages II–IV together take 79% of the total wall-clock, and the remaining 21% is data loading and eval. Most of the win comes from Stage I seeding a good initialisation — once the experts are well-decomposed, RL needs far fewer samples to converge.

Conclusion

PADD recovers the latent module structure of a router-less dense teacher into a pretrained MoE student and learns stable routing — *without any router labels from the teacher.* The student **beats its own 7B teacher by 2.5 pt** on Qwen avg Pass@1, at 0.47× per-token compute, with broad-skill accuracy preserved.

Limitations & future work.

- PADD currently needs a same-family dense teacher; cross-family distillation is left for future work.
- Stage I assumes a sufficient corpus to surface co-activation structure; very small corpora may under-cluster.
- The benefit on non-reasoning tasks is smaller; PADD is best suited for math/code domains where the dense teacher has strong specialization.

[1]Intel Corporation, China | [2]Zhejiang University, China

xinyue.peng@intel.com | yi.qian@intel.com