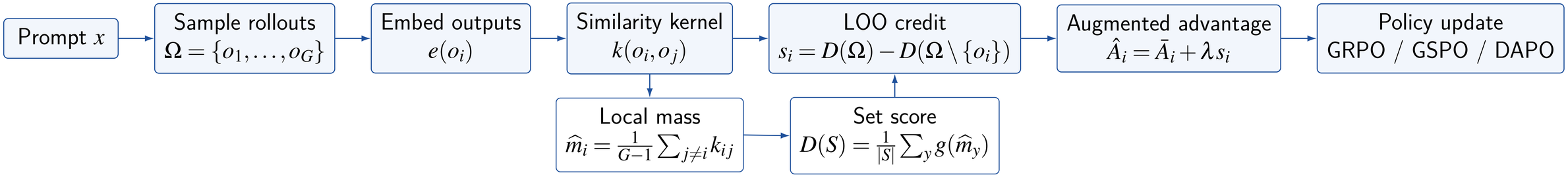
SetPO: Set-Level Policy Optimization for Diversity-Preserving LLM Reasoning

Chenyi Li[1,*] Yuan Zhang[2,*,†] Bo Wang[3] Guoqing Ma[2] Wei Tang[2] Haoyang Huang[2] Nan Duan[2]
[1]Peking University [2]JD Explore Academy, China [3]Beijing Institute of Technology
*Equal contribution †Project leader

Main Idea

Takeaway: SetPO assigns each rollout a leave-one-out set-diversity credit, then incorporates that credit into GRPO/GSPO/DAPO advantages to preserve diverse reasoning modes while improving accuracy.



1. Measure semantic crowding
Complete trajectories are compared in embedding space, so the signal captures solution-level novelty rather than surface-form entropy.

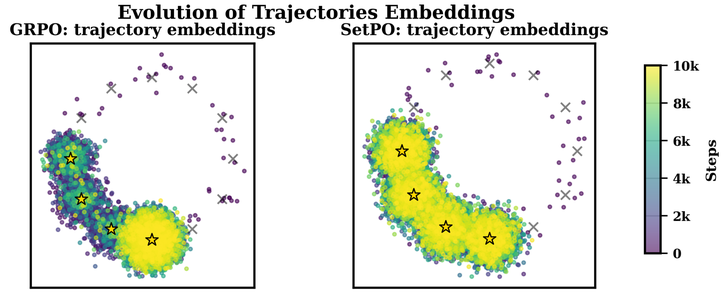
2. Assign marginal diversity
Removing each trajectory reveals whether it covers a distinct mode or mostly duplicates nearby samples.

3. Preserve plug-in compatibility
Only the advantage changes; clipping, KL control, and the base policy-optimization objective remain unchanged.

Motivation: RLVR Can Collapse Reasoning Modes

- RL with verifiable rewards improves mathematical reasoning, but its gains often concentrate probability mass on a narrow set of solutions.
- Token entropy is a noisy diversity proxy: it is length-biased and cannot distinguish semantically different complete solutions.
- Optimizing only multi-sample success can improve Pass@K without reliably improving single-sample correctness.
- SetPO directly rewards trajectories that add new semantic coverage to the sampled rollout set.

Core problem. Standard group-relative optimization ranks trajectories mainly by scalar correctness rewards. When several rollouts solve a prompt through similar reasoning, repeated high-reward samples reinforce the same mode and make alternative valid strategies harder to maintain during training.



Toy evidence. GRPO moves toward one dominant cluster, while SetPO keeps rollouts spread across multiple correct modes.

Set-Level Diversity Objective

Trajectory distribution. For a prompt x, a policy induces $P_\theta(y \mid x) = \prod_{t=1}^{T} \pi_\theta(a_t \mid x, a_{<t})$ over complete trajectories.

Kernelized local mass.

$$m_P(y) = \mathbb{E}_{y' \sim P}\left[k(y, y')\right], \qquad k(y, y') = \mathrm{sim}(e(y), e(y')) \in [0, 1].$$

Small local mass means the trajectory lies in a sparse semantic region.

Population diversity.

$$\mathscr{F}(P) = \mathbb{E}_{y \sim P}\left[g(m_P(y))\right], \qquad g(x) = -\log(1 + x).$$

Since g is non-increasing, sparse trajectories obtain larger value.

Monte Carlo set score.

$$D(S) = \frac{1}{|S|}\sum_{y \in S} g\left(\frac{1}{|S| - 1}\sum_{z \in S \setminus \{y\}} k(y, z)\right).$$

Leave-one-out marginal.

$$s_i = D(\Omega) - D(\Omega \setminus \{o_i\}), \qquad \hat{A}_i = \bar{A}_i + \lambda s_i.$$

The result is a drop-in shaping term for group-based policy optimization.

Theory: Why the Credit Penalizes Redundancy

Population perturbation. For $P_\varepsilon = (1 - \varepsilon)P + \varepsilon\delta_\tau$, the influence of a trajectory decomposes into novelty and interaction:

$$\mathscr{I}(\tau; P) = g(m_P(\tau)) + \mathbb{E}_{y \sim P}\left[g'(m_P(y))k(y, \tau)\right] - \Psi(P).$$

The first term favors low local mass; the second penalizes similarity to existing regions.

Finite-set monotonicity. For any $t \neq i$,

$$\frac{\partial s_i}{\partial k(o_i, o_t)} = \frac{g'(m_i) + g'(m_t)}{|\Omega|(|\Omega| - 1)} < 0.$$

Increasing similarity to another rollout strictly lowers the marginal contribution. This creates the desired diminishing-returns signal: rarer trajectories receive more credit.

Interpretation. A trajectory receives high credit only when it occupies a sparse region and does not congest the neighborhoods of other rollouts. This makes SetPO different from pairwise repulsion: credit is assigned by contribution to the whole sampled set.

Main Results

53.8% best 7B avg Pass@1 — **+5.6** avg gain over GRPO at 7B — **<10%** extra wall-clock time

SetPO improves GRPO, GSPO, and DAPO across 1.5B/7B math benchmarks and scales to 32B.

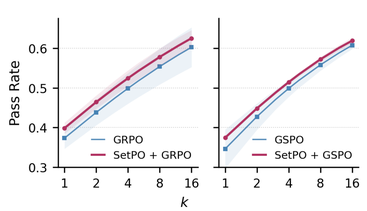
Mathematical Reasoning Benchmarks

Average Pass@1 across GSM8K, MATH500, College Math, AMC23, AIME24, and AIME25.

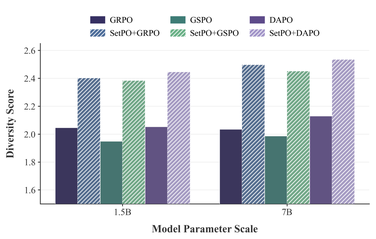
Scale	GRPO	GSPO	DAPO	SetPO+GRPO	SetPO+GSPO	SetPO+DAPO
1.5B	43.4	43.2	44.7	**46.7** (+3.3)	45.2 (+2.0)	**46.7** (+2.0)
7B	47.2	48.4	51.7	52.8 (+5.6)	51.1 (+2.7)	**53.8** (+2.1)

Hard benchmark gains at 7B.

Method	AMC23	AIME24	AIME25	College Math
GRPO	53.5	15.4	9.7	42.2
SetPO+GRPO	60.5	21.6	**13.6**	48.3
DAPO	59.6	21.7	11.0	47.1
SetPO+DAPO	**62.3**	**25.3**	13.5	**49.5**



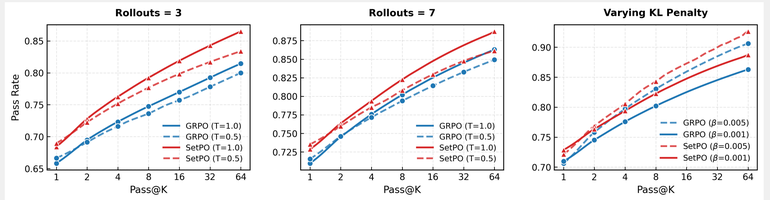
Olympiad. SetPO improves performance and reduces variance for both GRPO and GSPO.



Diversity. SetPO-enhanced methods obtain higher AIME24 solution-diversity scores.

Pass@K, Countdown, and Scaling

Countdown task. SetPO improves sample efficiency under different rollout counts, temperatures, and KL penalties.



Countdown diversity

Setting	Width	Mode
R7 GRPO	351	2.3
R7 SetPO	**415**	**3.5**
R3 GRPO	321	2.1
R3 SetPO	**403**	**3.2**

Reading. Wider solution coverage translates into better Pass@K: SetPO keeps useful alternatives alive instead of merely spreading token-level surface forms.

32B scaling. Representative Pass@K trends on Qwen2.5-32B.

Takeaway.

- SetPO is not a small-model artifact: the same set-level credit improves Pass@K at the 32B scale.
- Gains become clearer when the sampler has a larger budget, where diverse reasoning modes matter most.
- Accuracy and diversity rise together because SetPO preserves distinct correct solution paths instead of only sharpening one mode.

Practical implications.

- **When to use it:** prompts with many plausible solution routes, where vanilla RLVR tends to over-commit to the first high-reward mode.
- **Why it is lightweight:** SetPO changes only the advantage term; the rollout budget, verifier, KL control, and optimizer remain unchanged.
- **What to expect:** better sample efficiency at large K without sacrificing single-sample benchmark accuracy.