SPO++: Stream-Aligned Policy Optimization for Asynchronous Agentic RL

Kai Ruan[1,† *] **Jinghao Lin**[2,†] **Qianshan Wei**[3]
Ziqi Zhou[4] **Zihe Huang**[5]
[1]Gaoling School of Artificial Intelligence, Renmin University of China
[2]Independent Researcher [3]Institute of Automation, Chinese Academy of Sciences
[4]Duke University [5]Institute of Computing Technology, Chinese Academy of Sciences
[†]Equal contribution

Abstract

Group-relative reinforcement learning waits for sibling rollouts of the same prompt, which is costly for long and variable tool-use trajectories. Single-stream Policy Optimization (SPO) removes this dependency with a persistent prompt-level value estimate, but its recipe whitens one advantage per trajectory before optimizing a token-mean actor loss. We show that trajectory centering generally does not center the token-weighted quantity consumed by the actor, and fix the mismatch by standardizing terminal-outcome advantages under the action-token measure. We additionally organize prompt evidence by the policy event that generated it rather than learner receipt order. Across matched runs on ALFWorld at two model scales and on Math-TIR, SPO++ improves online learning efficiency over SPO. A paired ablation identifies action-token-measure normalization as the strongest tested component.

1 Introduction

Reinforcement learning with verifiable outcomes is increasingly used to train language-model reasoning and tool use. GRPO estimates a critic-free advantage from multiple outputs sampled for one prompt, and later large-scale recipe refinements improve the stability of this family [14, 19]. For agents, however, the group is also a barrier: environment interaction, tool calls, and failed trajectories create long completion-time tails, yet the update waits for the slowest sibling.

SPO replaces the ephemeral group baseline with persistent prompt-level evidence, enabling one online rollout per prompt [17]. This is attractive for asynchronous execution, but exposes two consistency questions. First, a sequential prompt tracker observes learner receipt order rather than the policy event that generated an outcome. Second, the original SPO recipe whitens equally weighted trajectory advantages and subsequently broadcasts each scalar over a variable number of response tokens. If the actor loss is averaged over tokens, response length silently changes the center seen by optimization.

We study these two mismatches in SPO++. The method retains SPO's terminal trajectory outcome and single-rollout dependency, but (i) freezes prompt values at dispatch and records returned evidence using a policy-event coordinate, and (ii) normalizes under the same action-token measure consumed by the loss. Together, these changes align the persistent baseline with the policy clock and the scalar advantage with the actor-loss measure.

Our contributions are:

*Corresponding author: kairuan@ruc.edu.cn.

Preprint. Work in progress.

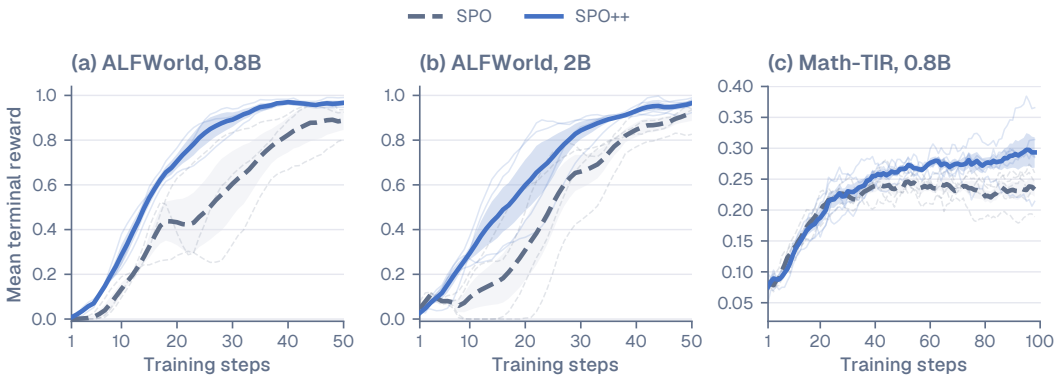

Figure 1: SPO++ acquires reward faster than SPO. Thin lines show matched replicates, thick lines their mean, and shaded regions one standard error. Curves use a five-step trailing mean; Table 1 reports unsmoothed aggregates.

- We identify a measure mismatch in trajectory whitening followed by a token-mean actor loss, and give action-token-measure normalization that preserves reward and credit granularity.
- We formulate a dispatch-causal, event-time prompt tracker whose estimate is invariant to receipt order for a fixed visible history.
- We evaluate SPO++ across two ALFWorld model scales and Math-TIR, and isolate the normalization measure in a matched ablation.

2 Method

Overview. SPO++ keeps SPO's one-rollout dependency graph and terminal outcome unchanged. A request reads and freezes a prompt baseline at dispatch; when it returns, its outcome is attached to the policy event that generated it. The resulting scalar advantages are standardized under the action-token measure and then consumed by the same token-mean actor loss. The two modifications therefore act at different boundaries: event-time memory governs which historical evidence defines the baseline, whereas measure alignment governs how a completed batch is standardized for optimization.

2.1 Single-stream prompt values

Let x denote a repeatable task identity, τ_i a completed trajectory, and $R_i \in \{0, 1\}$ its terminal outcome. SPO maintains success and failure evidence (α_x, β_x) and reads a pre-update prompt value

$$\widehat{v}_x = \frac{\alpha_x}{\alpha_x + \beta_x}, \qquad A_i = R_i - \widehat{v}_x. \tag{1}$$

After consumption, evidence is updated with a policy-drift-dependent retention factor ρ_i:

$$\alpha_x \leftarrow \rho_i \alpha_x + R_i, \qquad \beta_x \leftarrow \rho_i \beta_x + (1 - R_i). \tag{2}$$

Each actor update uses one online rollout per prompt, while persistent evidence supplies its prompt baseline. This single-stream dependency graph distinguishes SPO from group-based estimators.

2.2 Event-time prompt memory

Sequentially applying Equation 2 in completion order makes the tracker depend on systems timing. A *policy event* is the published actor snapshot used to generate a request. Let z_i denote the coordinate of the snapshot that generated trajectory i; successive snapshots advance this coordinate by the configured decay interval. The pair (α_0, β_0) is the frozen offline evidence initialized at coordinate z_0. SPO++ freezes $\widehat{v}_x$ when a request is dispatched. For a future dispatch q, let $\mathcal{H}_q(x)$ contain

exactly the outcomes for task x visible before that dispatch. We form

$$\alpha_q = e^{-(z_q - z_0)}\alpha_0 + \sum_{i \in \mathcal{H}_q(x)} e^{-(z_q - z_i)} R_i,$$
$$\beta_q = e^{-(z_q - z_0)}\beta_0 + \sum_{i \in \mathcal{H}_q(x)} e^{-(z_q - z_i)}(1 - R_i), \qquad \widehat{v}_q = \frac{\alpha_q}{\alpha_q + \beta_q}. \tag{3}$$

Remark 1 (Receipt-order invariance)**.** *For fixed q, $\mathcal{H}_q(x)$, and event coordinates, Equation 3 is invariant to receipt permutation up to floating-point reduction error: reordering receipts only permutes two finite sums. Receipt-order invariance is therefore conditional on the fixed visible history $\mathcal{H}_q(x)$.*

A returned outcome updates the baseline for subsequent dispatches; its own baseline remains the snapshot read at dispatch.

Implementation regime. The experiments dispatch a new request for a prompt after its previous request returns, while retaining first-finished completion across different prompts. For each generation-policy event, the implementation advances z by $-\log \rho$ with $\rho = 0.875$; an outcome generated k policy events before dispatch therefore receives weight ρ^k. Equation 3 is the event-indexed closed form of geometric retention, whereas the SPO baseline derives retention from post-update token drift.

2.3 Action-token-measure normalization

Let n be the number of trajectories in a batch and L_i the number of valid generated action tokens in trajectory i; observation and padding tokens have zero weight. Trajectory-wise whitening defines

$$\mu_{\text{traj}} = \frac{1}{n}\sum_i A_i, \qquad \sigma^2_{\text{traj}} = \frac{1}{n-1}\sum_i (A_i - \mu_{\text{traj}})^2, \qquad \widetilde{A}^{\text{traj}}_i = \frac{A_i - \mu_{\text{traj}}}{\sigma_{\text{traj}} + 10^{-8}}. \tag{4}$$

It guarantees $\sum_i \widetilde{A}^{\text{traj}}_i = 0$, but a token-mean loss sees

$$\frac{\sum_i L_i \widetilde{A}^{\text{traj}}_i}{\sum_i L_i} = \frac{\text{Cov}_n(L, \widetilde{A}^{\text{traj}})}{\overline{L}}, \tag{5}$$

where $\overline{L} = n^{-1}\sum_i L_i$ and Cov_n uses divisor n. The residual is therefore generally nonzero when token count and advantage co-vary. SPO++ instead standardizes under the same action-token measure consumed by the loss:

$$\mu_{\text{tok}} = \frac{\sum_i L_i A_i}{\sum_i L_i}, \qquad \sigma^2_{\text{tok}} = \frac{\sum_i L_i (A_i - \mu_{\text{tok}})^2}{\sum_i L_i - 1}, \qquad \widetilde{A}^{\text{tok}}_i = \frac{A_i - \mu_{\text{tok}}}{\sigma_{\text{tok}} + 10^{-8}}. \tag{6}$$

Every valid action token in one trajectory receives the same scalar $\widetilde{A}^{\text{tok}}_i$. The modification changes the normalization measure while preserving trajectory-level credit granularity.

2.4 Optimization and asynchronous execution

Both methods use the same uncertainty sampler, with prompt weight $\sqrt{\widehat{v}_x(1-\widehat{v}_x)} + 0.05$, and collect the first completed requests within each policy version. Old-policy work is drained when new parameters are published, and rollout importance ratios use an upper truncation threshold of two. For negative advantage A, Dual-Clip caps the per-token surrogate loss at $-cA$; the evaluated SPO and SPO++ recipes use $c = 10$ and $c = 2$, respectively. Auxiliary reward, KL regularization, and moving-reference updates are disabled in both recipes. Appendix A reports the remaining implementation details.

3 Experiments

3.1 Protocol and metrics

We compare SPO and SPO++ in matched independent runs with identical model checkpoints, prompt sets, frozen offline prompt-value initializations, decoding, trajectory budgets, optimizer settings, and asynchronous runtime. One plotted ALFWorld step averages 128 trajectories; one plotted

Math-TIR step averages two 128-trajectory updates. Thus 50 ALFWorld steps and 100 Math-TIR steps correspond to 6,400 and 25,600 online trajectories per method, respectively. Our primary metric is normalized reward-curve area over the common budget: trapezoidal AUC for ALFWorld and the step mean (normalized rectangle-rule area) for Math-TIR. The definitions differ only in endpoint weighting and are numerically near-identical on these curves; we retain each domain's original comparator. Appendix A gives both definitions. The secondary metric is the mean reward over the final five plotted steps. Differences are paired within matched replicates and always denote SPO++ minus SPO; we report their mean and sample standard deviation.

Recipe variants. SPO++ combines event-time memory and action-token-measure normalization with fixed retention $\rho = 0.875$ and negative Dual-Clip $c = 2$. SPO uses drift-dependent retention and $c = 10$. Fixed retention gives policy events a learner-independent geometric coordinate. The end-to-end comparison therefore measures the full recipes; only the normalization measure is isolated in Table 2. Dual-Clip is not ablated.

ALFWorld. We evaluate Qwen3.5-0.8B and Qwen3.5-2B agents [12] on 128 canonical ALFWorld tasks [15]. The agent receives the complete AgentLoop interaction history, and each model scale uses its own frozen prompt-value initialization. Results are averaged over three matched runs at 0.8B and four at 2B; the task set, context, interaction limit, and online trajectory budget are otherwise identical.

Math-TIR. We evaluate Qwen3.5-0.8B with native Python tool use on a 1,500-example training split of DAPO-Math-17K [19]. Both methods share the same supervised cold-start checkpoint and prompt-value initialization. Results are averaged over five matched runs. Appendix A gives the full protocol.

3.2 Main results and training dynamics

Table 1: SPO++ improves online learning efficiency in every evaluated setting. Paired differences are reported in percentage points on the normalized $[0, 1]$ scale, as mean $\pm$ sample s.d.; end reward averages the final five plotted steps.

	ALFWorld		Math-TIR
	Qwen3.5-0.8B	Qwen3.5-2B	Qwen3.5-0.8B
SPO area	0.532	0.522	0.217
SPO++ area	**0.722**	**0.681**	**0.242**
Δ area (points)	$+19.00 \pm 8.95$	$+15.92 \pm 10.25$	$+2.50 \pm 1.64$
Δ end reward (points)	$+7.86 \pm 7.18$	$+4.88 \pm 6.83$	$+5.03 \pm 4.74$

Overall performance and learning efficiency. Table 1 and Figure 1 show a consistent advantage across model scales and task families. Every ALFWorld run improves in curve area, and six of seven also finish with a positive end-reward difference. Math-TIR shows a smaller positive mean gain, with both metrics improving in four of five runs. The matched curves begin together and separate during training in all three settings.

When does measure mismatch matter? Equation 5 predicts that trajectory whitening departs most from the actor's measure when response token count co-varies with advantage. The larger gains on long-horizon ALFWorld and the smaller Math-TIR improvement are qualitatively consistent with this prediction.

3.3 Ablation and optimization diagnostics

Action-token-measure normalization. We isolate the normalization measure while holding the prompt tracker, retention, Dual-Clip, importance correction, optimizer, and the presence of standardization fixed. Table 2 shows that action-token-measure normalization raises the short-horizon

learning composite by 10.70 ± 3.98 percentage points relative to the otherwise matched trajectory-normalized variant. Equation 6 sets the actor-weighted mean $\sum_i L_i \widetilde{A}_i^{\mathrm{tok}} / \sum_i L_i$ to zero by construction.

Optimization diagnostics. Figure 2 compares the complete SPO and SPO++ recipes. SPO++ exhibits lower PPO KL and clip fraction at a comparable gradient-norm scale. Because retention and negative Dual-Clip also differ, these curves are recipe-level diagnostics rather than an isolated normalization effect.

Figure 2: Recipe-level optimization diagnostics for SPO and SPO++. Curves show one matched Math-TIR run with complete per-update logging; pale traces are raw updates and dark curves are five-step trailing means. Other runs record a reduced diagnostic subset.

Table 2: Matched normalization ablation on 0.8B ALFWorld over ten training steps and three seeds. Values are changes in the learning composite C_{10}, in percentage points; the first two rows are relative to SPO and the last row is the paired difference. We report mean $\pm$ sample s.d.

Variant	Normalization measure	Δ learning composite
SPO++ (trajectory norm.)	Trajectory	$+2.29 \pm 9.66$
SPO++	Action token	$\mathbf{+12.99 \pm 8.21}$
Action-token vs. trajectory	Paired difference	$\mathbf{+10.70 \pm 3.98}$

4 Related Work

Single-rollout advantage estimation. PPO introduced clipped surrogate policy optimization [13]. GRPO and subsequent large-scale refinements replace the critic with within-prompt relative comparison [14, 19]; SPO instead carries prompt-level evidence across visits [17]. Other single-rollout estimators obtain a baseline by sharing information across different prompts [2], while MSSR stabilizes multimodal single-rollout RL through entropy-based advantage shaping [9]. Our work preserves SPO's persistent empirical prompt baseline and studies two consistency properties: matching standardization to the actor-loss measure and separating policy event time from learner receipt time.

Learned and implicit values. Value-based approaches improve advantage estimation by changing the source of the baseline. VC-PPO pretrains a critic and decouples actor and critic GAE targets [20], while VAPO adds length-adaptive GAE and long-reasoning stabilizers [21]. V_0 predicts state-zero success with a frozen generalist value model [25], and $V_{0.5}$ combines this prior with sparse empirical rollouts and adaptive sampling [24]. SAO and SAPO likewise learn values for single-response optimization, using separate and shared actor–value architectures, respectively [4, 7]; BPCO instead stabilizes a bounded critic with Monte Carlo targets and retains raw residual advantages [11].

Objectives and supervision granularity. Other methods alter the objective or the granularity of supervision. MaxRL moves the terminal-binary objective toward a compute-indexed likelihood

approximation [16]; VIMPO derives token-level value recurrences from policy–reference log-ratios [5]; and SRPO turns hindsight reflections into dense token-level distillation targets [8]. Token-level loss aggregation is also an explicit stability choice in large-scale group-relative training [19]. Dr. GRPO identifies response-level length bias in GRPO [10], while Balanced Aggregation analyzes sign–length coupling induced by group-relative aggregation rules [22]. These works change how response or token terms are aggregated. Our change holds the token-mean loss and terminal reward fixed, and instead matches the measure used to standardize SPO's persistent scalar advantages. The two design axes compose; SPO++ remains critic-free and preserves the expected-reward objective.

Asynchronous policy optimization. Asynchronous LLM RL systems decouple rollout and optimization but must manage policy lag [1]. PNPO reuses stale rollouts through prefix-aware ratios [23], A-3PO uses staleness-aware anchors [6], and recent analyses expose old-policy mismatch and adapt trust regions [3, 18]. We therefore separate policy event time, learner receipt time, and behavior-policy correction. We evaluate on text-based ALFWorld and SPO's rule-verified TIR setting [15, 17, 19].

5 Limitations

Our experiments study online learning efficiency on small Qwen3.5 models under limited training budgets; future evaluations can extend to longer horizons and out-of-distribution tasks. The end-to-end comparison changes event-time tracking, retention, and Dual-Clip jointly, while the shorter 0.8B ALFWorld ablation isolates normalization under standardized advantages. That ablation changes both centering and batchwise scaling, and does not decompose their individual effects. The one-request same-prompt concurrency cap restricts event-time experiments to cross-prompt completion order. Persistent prompt values require repeatable task identities and offline initialization, first-completed collection can favor shorter trajectories, and outcome-level credit retains the sparse-reward cold-start challenge.

6 Conclusion

We introduced SPO++, which aligns single-stream advantage normalization with the action-token measure consumed by a token-mean actor loss and organizes persistent prompt evidence by policy event time. Across two ALFWorld model scales and Math-TIR, SPO++ improves reward-curve area over SPO. A paired ablation identifies action-token-measure normalization as the strongest isolated contributor. Single-stream RL couples dependency reduction to persistent statistics that must align with both the policy clock and the measure optimized by the actor.

References

[1] Wei Fu, Jiaxuan Gao, Xujie Shen, Chen Zhu, Zhiyu Mei, Chuyi He, Shusheng Xu, Guo Wei, Jun Mei, Jiashu Wang, Tongkai Yang, Binhang Yuan, and Yi Wu. AReaL: A large-scale asynchronous reinforcement learning system for language reasoning. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1218.

[2] Shijin Gong, Erhan Xu, Kai Ye, Francesco Quinzan, Giulia Livieri, and Chengchun Shi. BASIS: Batchwise advantage estimation from single-rollout information sharing for LLM reasoning. *arXiv preprint arXiv:2605.27293*, 2026.

[3] Zhong Guan, Yongjian Guo, Haoran Sun, Wen Huang, Shuai Di, Likang Wu, Xiong Jun Wu, and Hongke Zhao. Missing old logits in asynchronous agentic RL: Semantic mismatch and repair methods for off-policy correction. *arXiv preprint arXiv:2605.12070*, 2026.

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

[5] Zhewei Kang, Aosong Feng, Sergey Levine, Dawn Song, and Xuandong Zhao. VIMPO: Value-implicit policy optimization for LLMs. *arXiv preprint arXiv:2606.20008*, 2026.

[6] Xiaocan Li, Shiliang Wu, and Zheng Shen. A-3PO: Accelerating asynchronous LLM training with staleness-aware proximal policy approximation. *arXiv preprint arXiv:2512.06547*, 2025.

[7] Dayang Liang, Lang Feng, Bo An, and Yunlong Liu. SAPO: Single-rollout autoregressive policy optimization for agentic reinforcement learning. *arXiv preprint arXiv:2608.19842*, 2026.

[8] Jialong Liu, Yuling Shi, Ning Yang, Xiaodong Gu, and Zuchao Li. SRPO: Self-reflective policy optimization for long-horizon reasoning. *arXiv preprint arXiv:2608.23493*, 2026. Accepted to ICML 2026.

[9] Rui Liu, Dian Yu, Lei Ke, Haolin Liu, Yujun Zhou, Zhenwen Liang, Haitao Mi, Pratap Tokekar, and Dong Yu. Stable and efficient single-rollout RL for multimodal reasoning. *arXiv preprint arXiv:2512.18215*, 2025.

[10] Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

[11] Penghui Qi, Xiangxin Zhou, and Wee Sun Lee. How to train a critic stably and efficiently. *arXiv preprint arXiv:2608.23566*, 2026.

[12] Qwen Team. Qwen3.5 model collection. Hugging Face, 2026. URL https://huggingface.co/collections/Qwen/qwen35.

[13] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. doi: 10.48550/arXiv.1707.06347.

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

[15] Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. ALFWorld: Aligning text and embodied environments for interactive learning. In *International Conference on Learning Representations*, 2021. arXiv:2010.03768.

[16] Fahim Tajwar, Guanning Zeng, Yueer Zhou, Yuda Song, Daman Arora, Yiding Jiang, Jeff Schneider, Ruslan Salakhutdinov, Haiwen Feng, and Andrea Zanette. Maximum likelihood reinforcement learning. *arXiv preprint arXiv:2602.02710*, 2026.

[17] Zhongwen Xu and Zihan Ding. Single-stream policy optimization. In *International Conference on Learning Representations*, 2026. Originally released as arXiv:2509.13232.

[18] Junyao Yang, Yucheng Shi, Zongxia Li, Zhongzhi Li, Ruhan Wang, Xiangxin Zhou, Kishan Panaganti, Haitao Mi, and Leowei Liang. Stale but stable: Staleness-adaptive trust regions for stabilizing asynchronous reinforcement learning. *arXiv preprint arXiv:2607.18722*, 2026.

[19] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, et al. DAPO: An open-source LLM reinforcement learning system at scale. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-3775. Originally released as arXiv:2503.14476.

[20] Yufeng Yuan, Yu Yue, Ruofei Zhu, Tiantian Fan, and Lin Yan. What's behind PPO's collapse in long-CoT? value optimization holds the secret. *arXiv preprint arXiv:2503.01491*, 2025.

[21] Yu Yue, Yufeng Yuan, Qiying Yu, Xiaochen Zuo, Ruofei Zhu, Wenyuan Xu, Jiaze Chen, Chengyi Wang, Tiantian Fan, Zhengyin Du, Xiangpeng Wei, Xiangyu Yu, Gaohong Liu, Juncai Liu, Lingjun Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Ru Zhang, Xin Liu, Mingxuan Wang, Yonghui Wu, and Lin Yan. VAPO: Efficient and reliable reinforcement learning for advanced reasoning tasks. *arXiv preprint arXiv:2504.05118*, 2025.

[22] Zhiyuan Zeng, Jiameng Huang, Zhangyue Yin, Jiashuo Liu, Ziniu Li, Bingrui Li, Yuhao Wu, Yining Zheng, Ge Zhang, Wenhao Huang, and Xipeng Qiu. Balanced aggregation: Understanding and fixing aggregation bias in GRPO. *arXiv preprint arXiv:2605.04077*, 2026.

[23] Wenhao Zhang, Yibo Xie, Rui Wang, Jiahua Yang, Lei Jiang, Zibo Yang, Yawei Wang, Jiali Xu, jasperawang, Haoyang Long, Huan Xiong, and alantzhao. Reusing rollouts under policy lag: Prefix-normalized policy optimization for LLM reinforcement learning. *arXiv preprint arXiv:2608.01418*, 2026.

[24] Yi-Kai Zhang, Yueqing Sun, Hongyan Hao, Qi Gu, Xunliang Cai, De-Chuan Zhan, and Han-Jia Ye. $V_{0.5}$: Generalist value model as a prior for sparse RL rollouts. *arXiv preprint arXiv:2603.10848*, 2026.

[25] Yi-Kai Zhang, Zhiyuan Yao, Hongyan Hao, Yueqing Sun, Qi Gu, Hui Su, Xunliang Cai, De-Chuan Zhan, and Han-Jia Ye. V_0: A generalist value model for any policy at state zero. *arXiv preprint arXiv:2602.03584*, 2026.

A Additional experimental details

A.1 Metric definitions

For ALFWorld rewards $r_1, \ldots, r_T$, normalized trapezoidal area is

$$\mathrm{AUC}_{\mathrm{ALF}} = \begin{cases} r_1, & T = 1, \\ \dfrac{1}{T-1} \displaystyle\sum_{t=1}^{T-1} \dfrac{r_t + r_{t+1}}{2}, & T > 1. \end{cases} \tag{7}$$

The Math-TIR comparator uses the step mean $T^{-1} \sum_t r_t$, which is the normalized rectangle-rule area. A matched comparison truncates both methods to their common number of completed reward points before computing either area or the final-five-collection mean. Main-text aggregates include replicates in which both methods reach the common training budget.

For the ten-step normalization ablation, the learning composite is

$$C_{10} = \frac{1}{2} \left(\mathrm{AUC}_{\mathrm{ALF}}(r_{1:10}) + \frac{1}{5} \sum_{t=6}^{10} r_t \right). \tag{8}$$

Thus, C_{10} equally weights normalized trapezoidal reward-curve area and the final-five reward mean, matching the frozen screening rule. We report C_{10} only for the ten-step normalization ablation; Table 1 reports reward-curve area and final-five reward separately.

A.2 Training and implementation details

Table 3: Training configuration for the reported experiments.

Item	ALFWorld	Math-TIR
Model	Qwen3.5-0.8B / 2B	Qwen3.5-0.8B (shared SFT)
Training prompts	128 canonical tasks	1,500 prompts
Offline value rollouts	1,024 (8 / prompt)	12,000 (8 / prompt)
Online trajectories / method	6,400	25,600
Update / publication	128 / every update	128 / every two updates
Train / rollout GPUs	2 / 6	4 / 4
Context / horizon	full AgentLoop / 50 interactions	full Python-tool / 8 turns
Decoding	$T = 1$, top-$p = 1$, top-$k = -1$	
PPO clip / epochs / LR / WD	$[0.2, 0.28]$ / 1 / 10^{-6} / 0.1	
KL / EWMA / entropy	0 / off / 0	
Reward shaping / prompt concurrency	none / 1	
Rollout IS	token upper-truncate at 2	

Across domains, SPO updates evidence in learner receipt order, derives retention from policy drift, whitens one scalar per trajectory, and uses negative Dual-Clip $c = 10$. SPO++ freezes values at dispatch, inserts returns by policy event time with $\rho = 0.875$, whitens under the action-token measure, and uses $c = 2$. Table 3 summarizes the shared controls.

The Math-TIR cold start precedes online RL. We sample eight native Python-tool trajectories per prompt from a Qwen3.5-4B teacher ($T = 0.6$, top-$p = 0.95$, top-$k = 20$) and retain verifier-positive, structurally valid traces. The retained 362 prompts are split 326/36 (764/73 examples). We train all parameters for two bfloat16 epochs with global batch 8, learning rate 10^{-5}, weight decay 0.01, cosine decay, and maximum length 40,960. The SFT loss is applied to the final assistant message.

Offline prompt values are Jeffreys-smoothed estimates of eight binary verifier outcomes per prompt (total Beta mass eight). Each pair shares its initialization: the model-specific base policy for ALFWorld and the shared cold-start policy for Math-TIR. All eight outcomes enter the initialization. Aggregates retain finite pairs that reach their common budget.