ACTION CONTEXT GROUPING AND ECHO PENALIZATION FOR AGENT POLICY OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Stepwise group relative reinforcement learning has emerged as a computationally efficient paradigm for training Large Language Model (LLM) agents in interactive environments with multiple turns. However, relying solely on environmental observations for state grouping introduces severe credit assignment biases. In this paper, we identify two fundamental optimization bottlenecks in this paradigm centered on observations: (1) *spatial degeneracy*, where ambiguous observations confound structurally different states, and (2) *temporal blindness*, where discounted future returns do not explicitly quantify historical redundancy associated with repetitive behavioral loops (the "Echo Trap"). To overcome these limitations without triggering the sample sparsity inherent in dense history matching, we propose **Action Echo Policy Optimization (AEPO)**. At the step level, AEPO introduces Action Conditioned Spatial Disambiguation (ACSD), constructing a soft weighted baseline via action text similarities to precisely isolate spatial noise. At the trajectory level, AEPO incorporates Difficulty Aware Echo Regularization (DAER), a dynamically calibrated penalty that squeezes out trajectory redundancy while preserving necessary exploration. Under matched training and evaluation settings, AEPO improves overall success rates over GRPO, GiGPO, RAGEN (StarPO-S), and ProxMO on ALFWorld and WebShop. With a 1.5B model, AEPO achieves a 93.0% ALFWorld success rate and adds 5.60% to advantage-computation time relative to GiGPO. Additional Search-R1 experiments extend the evaluation to multi-turn retrieval with free-form queries.

1 INTRODUCTION

Reinforcement Learning (RL) has become essential for training Large Language Model (LLM) agents in complex interactive tasks (Schulman et al., 2017; Shridhar et al., 2020; Wang et al., 2023). Among various RL algorithms, Group Relative Policy Optimization (GRPO) (Shao et al., 2024), alongside other critic-free paradigms (Yu et al., 2026), has emerged as a scalable approach by computing advantages through group-based normalization without explicit value networks, demonstrating strong performance in single-turn scenarios such as mathematical reasoning (Guo et al., 2025).

Recent efforts have extended GRPO to real-world multi-turn interactive tasks (Wang et al., 2025; Feng et al., 2026), requiring agents to execute sequential decision-making processes over extended horizons (e.g., ALFWorld (Shridhar et al., 2020)). In such practical settings, agents confront a highly heterogeneous landscape where task difficulty varies dramatically (Chen et al., 2025), while managing observation spaces laden with high-dimensional and repetitive states.

First, in the *spatial dimension*, text-based environments frequently emit highly degenerate observations, i.e., structural "state black holes" (generic responses like `"Nothing happens"`), that completely mask true underlying task progress. Because these observations are inherently aliased, the true outcome of an agent's current action is critically sensitive to its historical intent. For instance, in ALFWorld, reaching "`Nothing happens`" via an investigative action ("`examine soapbar`") frequently yields ultimate task success, whereas reaching the exact same text via a redundant action ("`pick up soapbar`") typically results in failure. By blindly pooling all visits to such states into a singular baseline group, existing methods forcibly average these heterogeneous contexts. The empirical variance reduction in Figure 1(a) motivates incorporating preceding-action context into the baseline.

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.

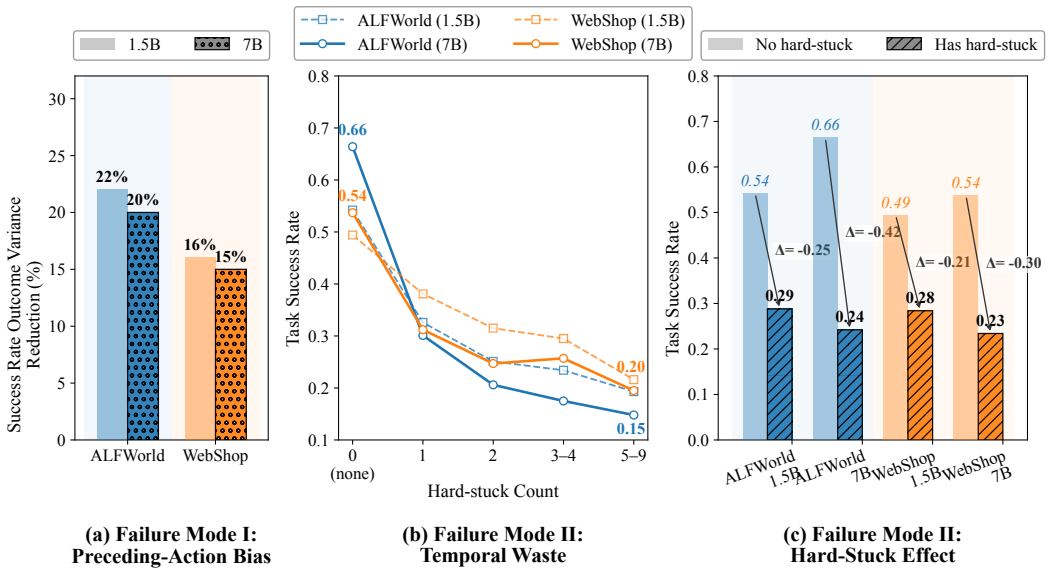

Figure 1: Empirical analysis of GiGPO failure modes: (a) Preceding-action bias represented by success rate variance reduction when conditioning on a_{t-1}; (b) Association between higher hard-stuck counts and lower task success rates; (c) Task success rate gap between trajectories with and without hard-stuck behavior, controlling for task difficulty through within-prompt comparisons.

Second, step-level methods such as GiGPO evaluate actions using discounted future returns, without explicitly quantifying the historical redundancy accumulated along a trajectory. Actions with identical subsequent reward sequences receive identical discounted returns, even when their preceding histories contain different amounts of repetition. Even after controlling for task difficulty, task success rates exhibit a pronounced overall decline as historical repetitions accumulate (Figures 1(b) and 1(c)). These observations motivate DAER, a trajectory-level regularizer that explicitly accounts for historical inefficiency and complements step-level credit assignment. Specifically, DAER uses arrival delay relative to successful rollouts as a proxy for historical inefficiency and weights this penalty by the empirical group success rate.

To resolve these spatial and temporal flaws without introducing the curse of dimensionality inherent in dense history matching, we propose **Action Echo Policy Optimization (AEPO)**. AEPO is a critic-free, low-overhead framework that combines two complementary mechanisms for multi-turn agent credit assignment. At the micro level, AEPO introduces *Action Conditioned Spatial Disambiguation (ACSD)* at the step level. Instead of explicitly fragmenting the sample pool into sparse subgroups, ACSD implements an attention-like, soft-weighted baseline. By leveraging text distance similarities over term frequency inverse document frequency action representations to dynamically allocate baseline weights, ACSD precisely isolates spatial noise while maintaining robust statistical smoothness. At the macro level, AEPO incorporates *Difficulty Aware Echo Regularization (DAER)* at the episode level. Calibrated dynamically by the batch success rate, DAER selectively penalizes accumulated temporal delay in optimized tasks to enforce path efficiency, while automatically vanishing in highly challenging environments to safeguard vital exploratory trial and error behaviors. Our core contributions are summarized as follows:

- We identify two limitations of observation-centric agent RL: preceding-action bias in state grouping and the absence of an explicit trajectory-level measure of historical inefficiency.
- We propose AEPO, a framework that combines step-level soft-weighted action conditioning (ACSD) with an episode-level difficulty-calibrated temporal efficiency penalty (DAER).
- AEPO improves overall success rates over strong RL baselines on ALFWorld and WebShop under matched settings, with additional gains on Search-R1 and low advantage-computation overhead.

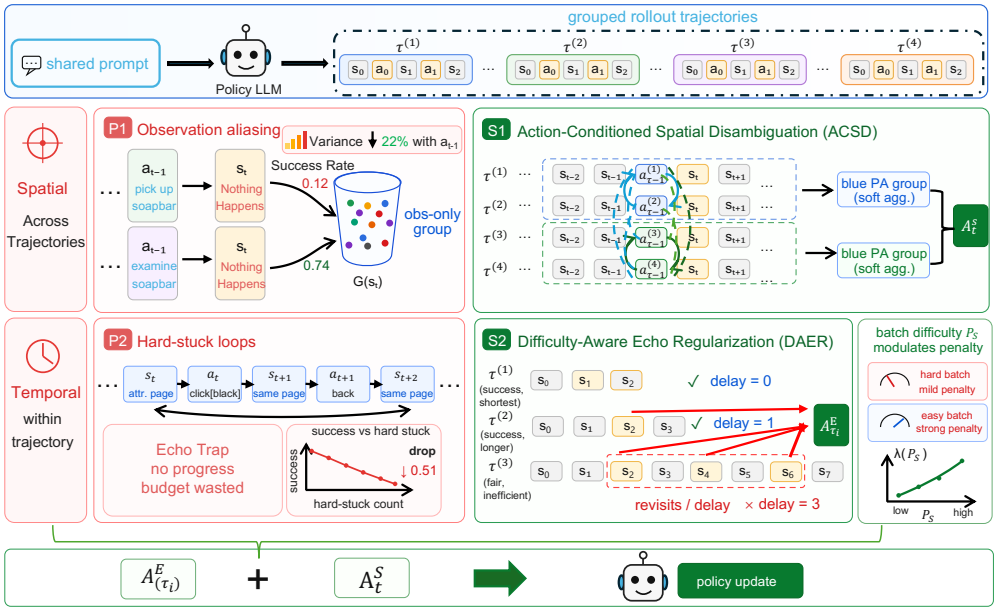

Figure 2: Overview of AEPO. Left: Two fundamental failure modes in observation-centric RL, namely (P1) spatial observation aliasing and (P2) temporal hard-stuck loops. Right: Our proposed solutions, (S1) Action-Conditioned Spatial Disambiguation (ACSD) for soft-weighted credit assignment and (S2) Difficulty-Aware Echo Regularization (DAER) for efficiency-adjusted trajectory optimization. The two components contribute step-level and trajectory-level guidance to policy updates.

2 Preliminaries

2.1 LLM Agents as POMDPs

We formulate the multi-turn interaction between a Large Language Model (LLM) agent and the environment as a Partially Observable Markov Decision Process (POMDP) (Sutton et al., 1998), formally defined by the tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{O}, \mathcal{Z}, \mathcal{R}, \gamma \rangle$.

At each discrete time step t, the agent receives a text-based observation $o_t \in \mathcal{O}$ generated by the emission function $\mathcal{Z}(o_t \mid s_t)$ from the underlying true environment state $s_t \in \mathcal{S}$. Conditioned on the historical context $x_t = (o_1, a_1, \ldots, o_t)$, the LLM policy $\pi_\theta(a_t \mid x_t)$ generates a textual action $a_t \in \mathcal{A}$. The environment then transitions to a subsequent hidden state $s_{t+1} \sim \mathcal{T}(\cdot \mid s_t, a_t)$ and yields a scalar reward $r_t = \mathcal{R}(s_t, a_t)$.

For ALFWorld and WebShop, the terminal task reward is 10 for success and 0 for failure, with an additional -0.1 penalty for each invalid action; other intermediate rewards are zero. Appendix B gives the task-specific reward scales. A complete interaction episode is defined as a rollout trajectory:

$$\tau = \{(o_1, a_1, r_1), (o_2, a_2, r_2), \ldots, (o_T, a_T, r_T)\} \tag{1}$$

where T denotes the maximum interaction horizon of the episode.

2.2 Group Relative Policy Optimization

To eliminate the heavy computational and memory footprints associated with training explicit value networks (Critics), Group Relative Policy Optimization (GRPO) (Shao et al., 2024) leverages intra-group statistics to estimate policy advantages.

For a given task description, the current policy samples a group of N outputs or complete trajectories $\{\tau_1, \tau_2, \ldots, \tau_N\}$ under identical initial environment states. GRPO computes the relative advantage

purely at the trajectory level:

$$A^E(\tau_i) = \frac{R(\tau_i) - \hat{\mu}_N}{\hat{\sigma}_N + \epsilon_{\text{norm}}} \tag{2}$$

where $R(\tau_i) = \sum_{t=1}^{T} r_t^{(i)}$ represents the holistic return of the trajectory τ_i, while $\hat{\mu}_N$ and $\hat{\sigma}_N$ denote the empirical mean and standard deviation of the returns across the N sampled trajectories within the group. $\epsilon_{\text{norm}} = 10^{-6}$ is the numerical-stability constant used in advantage normalization.

Crucially, this trajectory-wise formulation assigns an identical advantage value to every single step inside τ_i, thereby fundamentally overlooking the fine-grained, step-level credit assignment required for multi-turn optimization.

2.3 Stepwise Credit Assignment via GiGPO

To achieve fine-grained credit assignment without relying on auxiliary value models, GiGPO (Feng et al., 2026) decouples rollout trajectories into individual steps and retroactively clusters them based on shared environment observations.

With rewards dominated by task completion, GiGPO utilizes the discounted Monte Carlo (MC) return to evaluate the long-term impact of an individual action $a_t^{(i)}$, aggregating all future rewards:

$$R_t^{(i)} = \sum_{k=0}^{T-t} \gamma^k r_{t+k}^{(i)} \tag{3}$$

where $\gamma \in (0, 1]$ is the temporal discount factor. Steps that share the same observation across the batch are aggregated into a step-level group $G^S(\tilde{o}) = \{(j, k) \mid o_k^{(j)} \approx \tilde{o}\}$, where $\approx$ denotes textual similarity matching. By backpropagating future outcomes, GiGPO estimates the step-level relative advantage by comparing alternative actions under the identical observation context:

$$A^S(a_t^{(i)}) = \frac{R_t^{(i)} - \hat{\mu}_{G^S(\tilde{o})}}{\hat{\sigma}_{G^S(\tilde{o})} + \epsilon_{\text{norm}}} \tag{4}$$

where $\hat{\mu}_{G^S(\tilde{o})}$ and $\hat{\sigma}_{G^S(\tilde{o})}$ are the empirical mean and standard deviation of the MC returns $\{R_k^{(j)}\}$ within the step-level cluster $G^S(\tilde{o})$. The final policy gradient in GiGPO is guided by a hierarchical combination of the trajectory-level advantage A^E and the step-level advantage A^S.

Limitations of Observation-Only Clustering. While GiGPO provides a granular advantage estimation, its reliance on isolated observations directly contradicts the POMDP formulation, introducing two critical vulnerabilities:

- **Preceding-Action Bias:** By clustering steps solely via o_t, GiGPO ignores the hidden state s_t and historical context x_t. Identical textual observations often stem from entirely different preceding intents a_{t-1}. This aggregates inherently distinct states, severely distorting the intra-group mean $\hat{\mu}_{G^S(\tilde{o})}$ and penalizing optimal transitions.
- **Historical Temporal Redundancy:** GiGPO evaluates action quality through the discounted MC return $R_t^{(i)}$, which summarizes subsequent rewards. The preceding interaction history can contain repeated actions or delayed progress that this return does not explicitly quantify.

These foundational bottlenecks motivate our proposed framework, which introduces Action-Conditioned Spatial Disambiguation (ACSD) and Difficulty-Aware Echo Regularization (DAER) to achieve contextually consistent and temporally efficient optimization.

3 Methodology: Action-Echo Policy Optimization (AEPO)

We propose Action-Echo Policy Optimization (AEPO), which combines action-conditioned step-level baselines with trajectory-level delay regularization. AEPO is a critic-free framework that combines two complementary mechanisms for credit assignment: resolving spatial ambiguity via historical *Actions*, and breaking continuous temporal redundancy by penalizing trajectory *Echoes*.

3.1 ACTION-CONDITIONED SPATIAL DISAMBIGUATION (ACSD)

Identical observations can occur in different interaction contexts and be associated with different subsequent returns. The immediately preceding action a_{t-1} provides a compact contextual cue for distinguishing these visits. ACSD incorporates this cue into the step-level baseline through preceding-action similarity weighting.

Instead of strictly partitioning the observation-centric group $\mathcal{G}(s_t)$ into fragmented hard subgroups—which severely exacerbates sample sparsity and destabilizes advantage estimation—we introduce a continuous, *soft-weighted baseline* mechanism. This approach dynamically reweights the baseline calculation for a target step i by assigning higher importance to intra-group steps j that share similar preceding causal intents.

Specifically, we leverage Term Frequency-Inverse Document Frequency (TF-IDF) to represent the actions. TF-IDF achieves competitive success rates with lower advantage-computation cost than dense action encoders (Appendix E). Let $\mathbf{v}(a)$ denote the L_2-normalized TF-IDF vector of an action string a. The semantic proximity between the preceding actions of target step i and any step $j \in \mathcal{G}(s_t)$ is measured via cosine similarity: $S_{ij} = \mathbf{v}(a_{t-1}^{(i)})^\top \mathbf{v}(a_{t-1}^{(j)})$.

To compute the action-conditioned baseline, we apply a softmax with temperature $\tau > 0$ to these similarities. All steps in the observation group participate, with weights given by:

$$w_{ij} = \frac{\exp(S_{ij}/\tau)}{\sum_{k \in \mathcal{G}(s_t)} \exp(S_{ik}/\tau)} \tag{5}$$

A lower temperature τ concentrates more weight on similar preceding actions, while $\tau \to \infty$ recovers the uniform observation-centric baseline of GiGPO.

Using these localized weights, we compute the action-aware baseline $\hat{\mu}_\tau^{(i)}$ for the Monte Carlo returns within $\mathcal{G}(s_t)$:

$$\hat{\mu}_\tau^{(i)} = \sum_{j \in \mathcal{G}(s_t)} w_{ij} R_j \tag{6}$$

We compute the step-level ACSD advantage as the difference between the Monte Carlo return and the action-conditioned weighted baseline:

$$A^{\mathrm{ACSD}}(a_t^{(i)}) = R_t^{(i)} - \hat{\mu}_\tau^{(i)} \tag{7}$$

For a singleton observation group, $w_{ii} = 1$ and the ACSD advantage is zero. More generally, if the preceding action of step i has strictly higher similarity to itself than to every other action in the group, $w_{ii} \to 1$ as $\tau \to 0^+$, and its advantage approaches zero.

3.2 DIFFICULTY-AWARE ECHO REGULARIZATION (DAER)

To explicitly account for historical redundancy in policy optimization, DAER penalizes trajectory-level temporal waste ("echoes") using successful rollouts as efficiency references, with the penalty calibrated by the empirical group success rate.

We quantify the redundancy of trajectory τ_i through its mean temporal delay $D_i \equiv \bar{\Delta}t(\tau_i) = \frac{1}{T_i}\sum_{t=1}^{T_i} \max(0, t - t_{\min}(s_t^{(i)}))$. Here, $t_{\min}(s)$ is the earliest step state s is reached by any successful rollout within the same task group (unvisited states yield 0 delay to preserve exploration). For the N trajectories sampled for task q, let $\bar{D}_q = \frac{1}{N}\sum_{j=1}^{N} D_j$ denote the mean delay within the group. DAER adjusts the reward advantage according to each trajectory's deviation from this baseline:

$$B_i = A_i^E - \eta p_q (D_i - \bar{D}_q), \tag{8}$$

where p_q is the success fraction among the current N trajectories for task q, and η controls the relative weight of the efficiency term. Trajectories with below-average delay receive a positive adjustment, whereas those with above-average delay receive a negative adjustment. Centering removes delay components shared across the group. For any common delay offset c_q, $(D_i + c_q) - (\bar{D}_q + c_q) = D_i - \bar{D}_q$. Thus, common delay costs associated with necessary task

Table 1: Performance on ALFWorld and WebShop. RL training results averaged over 3 random seeds (mean ± std). **Bold**: best per section; underline: second best.

Method	ALFWorld							WebShop	
	Pick	Look	Clean	Heat	Cool	Pick2	**All**	**Score**	**Succ.**
Closed-Source LLMs									
GPT-4o	75.30	60.80	31.20	56.70	21.60	49.80	48.00	31.80	23.70
Gemini-2.5-Pro	92.80	63.30	62.10	69.00	26.60	58.70	60.30	42.50	35.90
Qwen2.5-1.5B-Instruct									
Origin	5.90	5.50	3.30	9.70	4.20	0.00	4.10	23.10	5.20
ReAct	17.40	20.50	15.70	6.20	7.70	2.00	12.80	40.10	11.30
Reflexion	37.80	24.00	23.30	14.50	20.70	3.90	23.50	55.80	21.90
PPO	77.42$_{\pm 8.06}$	64.71$_{\pm 1.85}$	39.29$_{\pm 1.52}$	66.67$_{\pm 5.66}$	65.00$_{\pm 5.16}$	50.00$_{\pm 4.94}$	60.16$_{\pm 1.70}$	77.02$_{\pm 2.78}$	55.47$_{\pm 2.55}$
GRPO	87.50$_{\pm 2.39}$	83.33$_{\pm 12.52}$	91.67$_{\pm 5.00}$	76.92$_{\pm 6.93}$	71.43$_{\pm 4.41}$	40.00$_{\pm 11.88}$	76.56$_{\pm 3.08}$	76.29$_{\pm 2.80}$	56.25$_{\pm 5.92}$
GiGPO	93.94$_{\pm 2.01}$	81.35$_{\pm 5.62}$	96.70$_{\pm 4.71}$	95.24$_{\pm 3.03}$	**81.82**$_{\pm 8.29}$	82.87$_{\pm 4.88}$	89.06$_{\pm 2.62}$	83.18$_{\pm 1.13}$	68.75$_{\pm 3.53}$
AEPO (Ours)	**94.90**$_{\pm 2.82}$	**87.04**$_{\pm 11.56}$	**100.00**$_{\pm 0.00}$	**100.00**$_{\pm 0.00}$	80.93$_{\pm 3.70}$	**95.26**$_{\pm 4.28}$	**92.97**$_{\pm 1.35}$	**84.05**$_{\pm 4.53}$	**71.88**$_{\pm 1.96}$
Δ *vs* GRPO	+8.5%	+4.5%	+9.1%	+30.0%	+13.3%	+138.2%	+21.4%	+10.2%	+27.8%
Qwen2.5-7B-Instruct									
Origin	34.80	22.90	18.10	7.30	2.50	3.60	16.20	26.40	7.80
ReAct	50.10	33.80	35.70	12.50	17.30	18.90	29.80	46.20	19.50
Reflexion	63.40	40.20	46.50	29.70	37.90	22.60	44.10	58.10	28.80
PPO	80.00$_{\pm 3.35}$	77.78$_{\pm 5.56}$	88.00$_{\pm 7.18}$	91.92$_{\pm 4.11}$	77.27$_{\pm 1.68}$	77.27$_{\pm 5.42}$	80.47$_{\pm 3.82}$	80.53$_{\pm 2.86}$	71.09$_{\pm 2.49}$
GRPO	89.50$_{\pm 8.35}$	66.67$_{\pm 0.00}$	91.90$_{\pm 7.33}$	78.31$_{\pm 7.15}$	73.13$_{\pm 6.67}$	64.96$_{\pm 2.96}$	75.78$_{\pm 1.22}$	82.79$_{\pm 3.30}$	72.66$_{\pm 2.08}$
GiGPO	96.29$_{\pm 3.21}$	84.07$_{\pm 4.49}$	93.75$_{\pm 0.00}$	90.11$_{\pm 8.58}$	**87.53**$_{\pm 4.52}$	88.89$_{\pm 5.09}$	92.19$_{\pm 3.80}$	83.35$_{\pm 3.17}$	75.78$_{\pm 3.83}$
AEPO (Ours)	**97.30**$_{\pm 2.78}$	**100.00**$_{\pm 0.00}$	**97.62**$_{\pm 4.12}$	**100.00**$_{\pm 0.00}$	85.30$_{\pm 4.00}$	**94.74**$_{\pm 1.73}$	**95.31**$_{\pm 1.36}$	**90.21**$_{\pm 3.96}$	**80.47**$_{\pm 2.86}$
Δ *vs* GRPO	+8.7%	+50.0%	+6.2%	+27.7%	+16.6%	+45.8%	+25.8%	+9.0%	+10.7%

execution cancel in the group comparison, and the efficiency adjustment depends on differences in delay across trajectories.

Since the reward advantages A_i^E have zero group mean, the adjusted advantages B_i retain this property. We then standardize the adjusted advantages to obtain the final DAER advantages:

$$A_i^{\mathrm{DAER}} = \frac{B_i}{\sigma_q(B) + \epsilon_{\mathrm{norm}}}, \tag{9}$$

where $\sigma_q(B)$ is the standard deviation of the adjusted advantages within the group and $\epsilon_{\mathrm{norm}} = 10^{-6}$ is the numerical-stability constant used in advantage normalization.

3.3 Unified AEPO Objective

AEPO combines trajectory-level and step-level guidance through

$$A_{\mathrm{AEPO}}(a_t) = A^{\mathrm{DAER}}(\tau) + \omega A^{\mathrm{ACSD}}(a_t) \tag{10}$$

DAER adjusts the trajectory-level reward advantage using arrival delay relative to successful rollouts. ACSD computes step-level advantages using a baseline weighted by preceding-action similarity. The coefficient ω controls the relative contribution of ACSD to the combined advantage, which is used in the clipped policy objective. Following the standard group-based optimization paradigm, we denote the probability ratio as $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\mathrm{old}}(a_t|s_t)}$. The overall policy objective maximizes the clipped likelihood ratio, fortified by a Kullback-Leibler divergence penalty to anchor the learned

policy against the reference model π_{ref}:

$$\mathcal{J}_{\text{AEPO}}(\theta) = \mathbb{E}\Bigg[\sum_{t=1}^{T}\Bigg(\min\Big(r_t(\theta)A_{\text{AEPO}}(a_t), \text{clip}(r_t(\theta), 1-\epsilon_{\text{clip}}, 1+\epsilon_{\text{clip}})A_{\text{AEPO}}(a_t)\Big) - \beta\mathbb{D}_{\text{KL}}(\pi_\theta\|\pi_{\text{ref}})\Bigg)\Bigg] \tag{11}$$

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Agentic Benchmarks. We evaluate our proposed AEPO on two widely adopted, real-world multi-turn interactive benchmarks: (1) **ALFWorld** (Shridhar et al., 2020): An embodied environment where the agent receives a text goal and must accomplish it through multi-turn interactions. It includes task instances across six categories of common household activities: Pick & Place (Pick), Examine in Light (Look), Clean & Place (Clean), Heat & Place (Heat), Cool & Place (Cool), and Pick Two & Place (Pick2). (2) **WebShop** (Yao et al., 2022a): A complex, web-based shopping environment designed to test LLM agents in realistic online shopping scenarios. Requiring HTML navigation and multi-step purchase decisions, it contains over 1.1 million products and 12K user instructions, providing a rich and diverse action space.

Baselines. To rigorously assess the effectiveness of AEPO, we compare it against a comprehensive suite of competitive baselines: (1) **Prompted reference models:** GPT-4o (Achiam et al., 2023) and Gemini-2.5-Pro (Team et al., 2023), used through prompting without task-specific parameter updates. (2) **Prompting agents:** ReAct (Yao et al., 2022b) and Reflexion (Shinn et al., 2023), which rely on in-context prompting to guide multi-step behavior without parameter updates. (3) **RL training methods:** This includes the traditional actor-critic method PPO (Schulman et al., 2017), the trajectory-level critic-free method GRPO (Shao et al., 2024), and the state-of-the-art stepwise relative advantage estimator GiGPO (Feng et al., 2026).

Implementation Details. We adopt Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct (Yang et al., 2025) as our base models. To ensure a strictly fair comparison, all RL training methods share identical hyperparameter configurations and training environments. Specifically, the rollout group size N is set to 8. Each LLM agent is prompted to first generate a chain-of-thought rationale enclosed within `<think></think>` tags, followed by the explicit action enclosed within `<action></action>` tags. The maximum interaction step limit is set to 50 for ALFWorld and 30 for WebShop. For the RL results in Table 1, we report the mean and standard deviation over 3 random seeds.

AEPO retains the common training hyperparameters of standard stepwise GRPO. We use an ACSD temperature of $\tau = 0.1$ for softmax weighting over each observation group; the DAER coefficient η controls the temporal penalty. The group success rate is computed directly from the current rollouts.

4.2 MAIN RESULTS

AEPO achieves the highest overall success rates among the compared RL methods on ALFWorld and WebShop at both model scales (Table 1).

AEPO achieves the best overall performance. Table 1 shows a pronounced gap between prompting baselines (Origin, ReAct, Reflexion) and RL training methods, confirming that RL is crucial for long-horizon agentic tasks. Among the RL methods, AEPO consistently outperforms traditional trajectory-level optimization (GRPO) and the state-of-the-art stepwise baseline (GiGPO). For the Qwen2.5-1.5B-Instruct model, AEPO achieves an impressive 93.0% overall success rate on ALFWorld (a $+21.4\%$ relative improvement over GRPO) and 71.9% on WebShop (a $+27.8\%$ improvement). This superiority scales robustly to the Qwen2.5-7B-Instruct model, where AEPO reaches 95.3% on ALFWorld and 80.5% on WebShop, outperforming GiGPO by a solid margin.

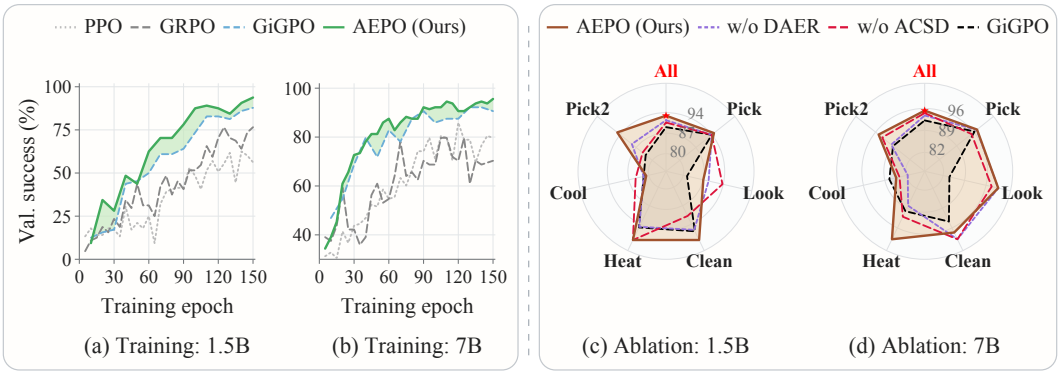

Figure 3: Training dynamics and component ablations on ALFWorld. (a,b) Success-rate curves for Qwen2.5-1.5B and Qwen2.5-7B, comparing AEPO with PPO, GRPO, and GiGPO. (c,d) Task-wise performance of AEPO, variants without ACSD or DAER, and GiGPO for the corresponding model sizes.

Comparison with recent RL methods. AEPO also achieves higher overall success rates than RAGEN (StarPO-S) and ProxMO on both benchmarks at both model scales. Detailed comparisons are provided in Appendix C.

Using Qwen2.5-3B-Instruct, AEPO achieves higher scores than GRPO and GiGPO on all four question-answering datasets in the Search-R1 setting (Appendix D).

Pronounced gains in complex, long-horizon scenarios. The performance improvements of AEPO are exceptionally pronounced in sub-tasks that require extended reasoning loops and are highly susceptible to temporal waste. For instance, on the ALFWorld "Look", "Heat", and "Pick2" tasks, which demand strict sequential dependency, GRPO struggles due to context-agnostic normalization. AEPO, however, achieves a perfect 100.0% success rate on "Heat" across both the 1.5B and 7B models, and reaches 100.0% on "Look" with the 7B model. Compared to GiGPO, AEPO's action-aware grouping and trajectory-level temporal penalty prevent the agent from falling into redundant loops, significantly accelerating convergence (as shown in the training curves in Figure 3(a,b)).

4.3 Ablation Study

Component Necessity and Synergy. Figure 3(c,d) illustrates the ablation results across distinct ALFWorld sub-tasks using a radar chart. Removing either component lowers the overall success rate. The components address different aspects of credit assignment. When Action-Conditioned Spatial Disambiguation is removed (AEPO w/o ACSD), performance drops significantly on complex interactive tasks like "Heat" and "Pick2". Removing DAER reduces the overall ALFWorld 1.5B success rate from 93.0% to 91.4%, a decrease of 1.6 percentage points, matching the $\eta = 0$ result in Figure 4(b). Difficulty scaling and group normalization further improve success rates on both benchmarks (Appendix F).

4.4 Hyperparameter Sensitivity

To further understand the robustness of AEPO, we evaluate its sensitivity to the core hyperparameters introduced in our framework. All sensitivity experiments use Qwen2.5-1.5B-Instruct on ALFWorld and WebShop.

Sensitivity to ACSD Temperature (τ). The temperature τ controls the concentration of the ACSD weights. As shown in Figure 4(a), performance peaks at $\tau = 0.1$, with success rates of 93.0% on ALFWorld and 71.9% on WebShop. A larger temperature ($\tau = 0.5$) gives more weight to dissimilar preceding actions, moving the baseline toward uniform observation-centric averaging. Smaller temperatures concentrate weight on fewer actions, reducing the effective sample size. The $\tau = 0$ reference uses exact preceding-action matching and uniformly averages the returns of matching steps; its WebShop success rate is 69.5%.

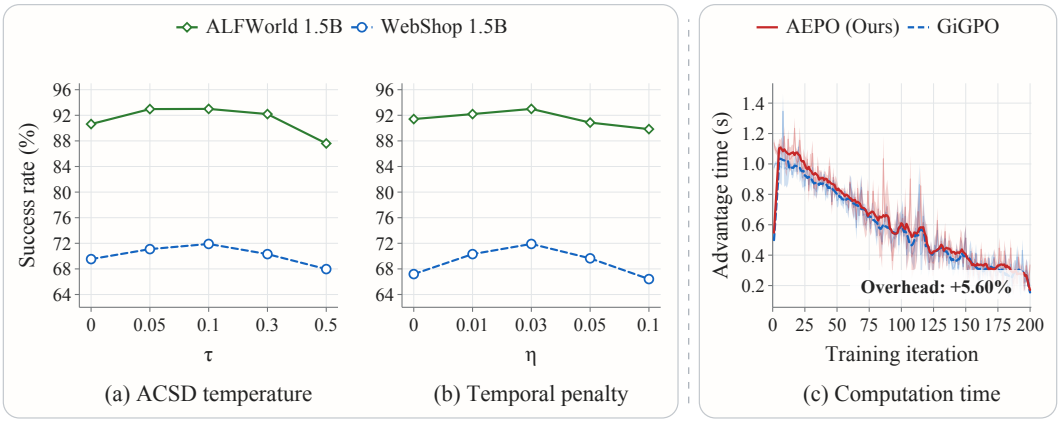

Figure 4: Hyperparameter sensitivity and computational efficiency. (a) Sensitivity to the ACSD temperature τ with $\eta = 0.03$; $\tau = 0$ denotes exact preceding-action matching. (b) Sensitivity to the temporal penalty η with $\tau = 0.1$. Sensitivity is evaluated with Qwen2.5-1.5B-Instruct on ALFWorld and WebShop. (c) Advantage computation time of AEPO and GiGPO; the mean over the 200 plotted iterations gives approximately 5.60% overhead (0.03272 s/iteration).

Sensitivity to Temporal Penalty Coefficient (η). The coefficient η governs the magnitude of the trajectory-level efficiency penalty. Figure 4(b) demonstrates optimal performance at $\eta = 0.03$. With the temporal penalty disabled ($\eta = 0$), success rates decrease to 67.2% on WebShop and 91.4% on ALFWorld. Increasing the penalty to $\eta = 0.1$ reduces WebShop success to 66.4%.

4.5 Computational Efficiency

AEPO introduces minimal computational overhead by adding only lightweight grouping and temporal scoring operations. Averaging the 200 iterations shown in Figure 4(c) gives an advantage-computation overhead of approximately $+5.60\%$ ($+0.03272$ s/iter) relative to GiGPO. Because advantage computation accounts for merely $\sim 0.17\%$ of the total training time, the net impact on end-to-end training efficiency is practically negligible.

5 Conclusion

In this paper, we propose AEPO, a robust framework designed to systematically address the fundamental spatial and temporal credit assignment bottlenecks in multi-turn reinforcement learning for LLM agents. We first demonstrate that relying strictly on isolated environmental observations precipitates severe preceding-action bias and traps agents in redundant behavioral loops. Subsequently, by decoupling the credit assignment process, we derive a principled method featuring two complementary mechanisms. Action-Conditioned Spatial Disambiguation (ACSD) mitigates spatial degeneracy by replacing hard boundaries with a soft-weighted baseline proportional to causal intent similarity. Concurrently, Difficulty-Aware Echo Regularization (DAER) adaptively penalizes trajectory delay based on task difficulty, thereby squeezing out temporal waste while preserving essential exploration. Extensive experiments across complex interactive benchmarks and open-weight models of varying scales demonstrate that our framework achieves consistent state-of-the-art improvements compared to strong baselines.

Limitations

Our evaluation covers Qwen2.5 models at 1.5B and 7B on ALFWorld and WebShop, and at 3B in the Search-R1 setting. These environments span structured text actions and free-form retrieval queries. Extending AEPO to other model families and larger models, as well as continuous actions and multimodal observations, is a direction for future work. Such settings call for suitable action representations and observation-matching mechanisms.

AI USE STATEMENT

Generative AI tools were used to assist with language polishing of the manuscript and the implementation of portions of the code.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting reinforce-style optimization for learning from human feedback in llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12248–12267, 2024.

Yuhan Chen, Yuxuan Liu, Long Zhang, Pengzhi Gao, Jian Luan, and Wei Liu. Step: Success-rate-aware trajectory-efficient policy optimization. *arXiv preprint arXiv:2511.13091*, 2025.

Yangyi Fang, Jiaye Lin, Xiaoliang Fu, Cong Qin, Haolin Shi, Chang Liu, and Peilin Zhao. Proximity-based multi-turn optimization: Practical credit assignment for llm agent training. *arXiv preprint arXiv:2602.19225*, 2026.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. Group-in-group policy optimization for llm agent training. *Advances in Neural Information Processing Systems*, 38:46375–46408, 2026.

Hiroki Furuta, Kuang-Huei Lee, Ofir Nachum, Yutaka Matsuo, Aleksandra Faust, Shixiang Gu, and Izzeddin Gur. Multimodal web navigation with instruction-finetuned foundation models. In *International Conference on Learning Representations*, volume 2024, pp. 29528–29558, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

Wouter Kool, Herke van Hoof, and Max Welling. Buy 4 reinforce samples, get a baseline for free!

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems*, 36:68539–68551, 2023.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652, 2023.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. Alfworld: Aligning text and embodied environments for interactive learning. *arXiv preprint arXiv:2010.03768*, 2020.

Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*, 2023.

Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, et al. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744–20757, 2022a.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022b.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026.

Zhehao Zhang, Weicheng Ma, and Soroush Vosoughi. Is gpt-4v (ision) all you need for automating academic data visualization? exploring vision-language models' capability in reproducing academic charts. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 8271–8288, 2024.

Zhuosheng Zhang and Aston Zhang. You only look at screens: Multimodal chain-of-action agents. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 3132–3149, 2024.

A Complete Mathematical Statements and Proofs

This appendix characterizes the population residual variance under action conditioning, the conditional estimation error of the ACSD baseline, and the centering and rank-preservation properties of DAER. It also describes how the trajectory-level and step-level advantages are combined in AEPO.

A.1 Formal Setup

We formulate the multi-turn interaction as a POMDP

$$\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{O}, \mathcal{Z}, \mathcal{R}, \gamma \rangle.$$

At step t, the environment is in a latent state s_t, while the agent only observes a textual observation $o_t \sim \mathcal{Z}(\cdot \mid s_t)$. Given history $x_t = (o_1, a_1, \ldots, o_t)$, the policy samples $a_t \sim \pi_\theta(\cdot \mid x_t)$. The environment then transitions by $s_{t+1} \sim \mathcal{T}(\cdot \mid s_t, a_t)$ and emits reward $r_t = \mathcal{R}(s_t, a_t)$. The discounted Monte Carlo return of step t in trajectory τ_i is

$$R_t^{(i)} = \sum_{k=0}^{T_i - t} \gamma^k r_{t+k}^{(i)}. \tag{12}$$

Definition 1: State Aliasing and Preceding-Action Bias. State aliasing occurs when different latent states can emit the same observation. Formally, there exist $s_A \neq s_B$ and o such that

$$\mathcal{Z}(o \mid s_A) > 0, \qquad \mathcal{Z}(o \mid s_B) > 0. \tag{13}$$

Preceding-action bias occurs when the latent posterior remains dependent on the immediately preceding action:

$$\begin{aligned} &\mathbb{P}(s_t = s_A \mid o_t = o, a_{t-1}) \\ &\quad \neq \mathbb{P}(s_t = s_A \mid o_t = o). \end{aligned} \tag{14}$$

For return-relevant aliased states, this means

$$I(a_{t-1}; R_t \mid o_t) > 0. \tag{15}$$

Thus, observation-only grouping incorrectly treats steps with the same observation as exchangeable, although their downstream returns may depend on the preceding action.

A.2 Population Residual Variance

Proposition 1: Residual Variance under Action Conditioning. Define the population residuals under observation-only and action-conditioned baselines as

$$\begin{aligned} A^{\mathrm{obs}} &= R_t - \mathbb{E}[R_t \mid o_t], \\ A^{\mathrm{cond}} &= R_t - \mathbb{E}[R_t \mid o_t, a_{t-1}]. \end{aligned} \tag{16}$$

Their conditional variances satisfy

$$\begin{aligned} &\mathrm{Var}(A^{\mathrm{obs}} \mid o_t) - \mathrm{Var}(A^{\mathrm{cond}} \mid o_t) \\ &\quad = \mathrm{Var}(\mathbb{E}[R_t \mid o_t, a_{t-1}] \mid o_t) \geq 0. \end{aligned} \tag{17}$$

The reduction in residual variance equals the variation in expected return explained by the preceding action within an observation group. It is strictly positive when the conditional expected return differs across preceding actions.

Proof. The action-conditioned residual has zero conditional mean given o_t and a_{t-1}. Applying the law of total variance to R_t and A^{cond} yields the stated identity. □

A.3 Soft ACSD Baseline

For a target step i, let $\mathcal{G}_i = \mathcal{G}(o_i)$ be the set of steps whose observations match o_i. ACSD represents each preceding action with an L_2-normalized TF-IDF vector $v(a)$. The action similarity is

$$S_{ij} = v(a_{t-1}^{(i)})^\top v(a_{t-1}^{(j)}). \tag{18}$$

The soft ACSD weight is

$$w_{ij} = \frac{\exp(S_{ij}/\tau)}{\sum_{k\in\mathcal{G}_i}\exp(S_{ik}/\tau)}, \qquad j \in \mathcal{G}_i, \tag{19}$$

where $\tau > 0$ is the temperature and the weights are computed over the entire observation group. The weighted baseline and step advantage are

$$\hat{\mu}_\tau^{(i)} = \sum_{j\in\mathcal{G}_i} w_{ij} R_j, \tag{20}$$

$$A^{\mathrm{ACSD}}(a_t^{(i)}) = R_t^{(i)} - \hat{\mu}_\tau^{(i)}. \tag{21}$$

Exact-Match Reference. For the $\tau = 0$ reference in Figure 4(a), let $\mathcal{G}_i^{\mathrm{exact}} = \{j \in \mathcal{G}_i : a_{t-1}^{(j)} = a_{t-1}^{(i)}\}$. The weights in this branch are

$$w_{ij}^{(0)} = \frac{\mathbf{1}\{j \in \mathcal{G}_i^{\mathrm{exact}}\}}{|\mathcal{G}_i^{\mathrm{exact}}|},$$

so the baseline is the mean return over steps with the same preceding action. The target step belongs to this set, ensuring a nonempty reference group.

Proposition 2: Error Decomposition for the ACSD Baseline. Let $Q(o,a) = \mathbb{E}[R_t \mid o_t = o, a_{t-1} = a]$. Let $\mathcal{F}_i$ contain the observations, preceding actions, and group membership used to construct the ACSD baseline for step i. Define the conditional return means and covariance matrix as

$$\begin{aligned} m_j &= \mathbb{E}[R_j \mid \mathcal{F}_i], \\ [\Sigma_i]_{jk} &= \mathrm{Cov}(R_j, R_k \mid \mathcal{F}_i). \end{aligned} \tag{22}$$

Let $\mathbf{m}_i = (m_j)_{j\in\mathcal{G}_i}$, $\mathbf{w}_i = (w_{ij})_{j\in\mathcal{G}_i}$, and $Q_i = Q(o_i, a_{t-1}^{(i)})$. The conditional mean squared error of the ACSD baseline satisfies

$$\begin{aligned} &\mathbb{E}\Big[(\hat{\mu}_\tau^{(i)} - Q_i)^2 \mid \mathcal{F}_i\Big] \\ &\qquad = \mathbf{w}_i^\top \Sigma_i \mathbf{w}_i + (\mathbf{w}_i^\top \mathbf{m}_i - Q_i)^2. \end{aligned} \tag{23}$$

The first term captures return variability and dependence among grouped steps. The second term measures the discrepancy between the weighted conditional mean and the target action-conditioned value.

Proof. Conditional on $\mathcal{F}_i$, the weights are fixed. Hence,

$$\begin{aligned} \mathbb{E}[\hat{\mu}_\tau^{(i)} \mid \mathcal{F}_i] &= \mathbf{w}_i^\top \mathbf{m}_i, \\ \mathrm{Var}(\hat{\mu}_\tau^{(i)} \mid \mathcal{F}_i) &= \mathbf{w}_i^\top \Sigma_i \mathbf{w}_i. \end{aligned} \tag{24}$$

The result follows from the bias–variance decomposition. □

A.4 DAER Stability

DAER penalizes temporally redundant trajectories at the episode level. Let $z_t^{(i)}$ be the normalized state key of step t in trajectory τ_i. For each state key z, define the earliest successful arrival time:

$$t_{\min}(z) = \min_{\tau_j : R(\tau_j) > 0} \min\{t : z_t^{(j)} = z\}. \tag{25}$$

If no successful trajectory reaches z, its delay contribution is set to zero. The mean delay of trajectory τ_i is

$$\bar{\Delta}t(\tau_i) = \frac{1}{T_i}\sum_{t=1}^{T_i} \max\left(0, t - t_{\min}(z_t^{(i)})\right). \tag{26}$$

For the current group of N trajectories sampled for the same task, the success rate is

$$p = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\{R(\tau_i) > 0\}. \tag{27}$$

Given the trajectory-level GRPO advantage $A^E(\tau_i)$, DAER first computes

$$\tilde{A}^{\mathrm{DAER}}(\tau_i) = A^E(\tau_i) - \eta p \bar{\Delta} t(\tau_i). \tag{28}$$

It then restores group-relative normalization:

$$A^{\mathrm{DAER}}(\tau_i) = \frac{\tilde{A}^{\mathrm{DAER}}(\tau_i) - \mu_{\mathrm{DAER}}}{\sigma_{\mathrm{DAER}} + \epsilon_{\mathrm{norm}}}, \tag{29}$$

where μ_{DAER} and σ_{DAER} are computed over the N trajectories in the group.

Theorem 3: Stabilization and Rank Preservation. DAER has zero empirical mean:

$$\frac{1}{N}\sum_{i=1}^{N} A^{\mathrm{DAER}}(\tau_i) = 0. \tag{30}$$

Its empirical variance is

$$\frac{\sigma^2_{\mathrm{DAER}}}{(\sigma_{\mathrm{DAER}} + \epsilon_{\mathrm{norm}})^2}. \tag{31}$$

Moreover, if two trajectories have equal trajectory advantage and

$$\bar{\Delta} t(\tau_i) < \bar{\Delta} t(\tau_j), \qquad A^E(\tau_i) = A^E(\tau_j),$$

then, whenever $\eta p > 0$,

$$A^{\mathrm{DAER}}(\tau_i) > A^{\mathrm{DAER}}(\tau_j). \tag{32}$$

Proof. The zero-mean property follows from subtracting the empirical mean:

$$\begin{aligned} \frac{1}{N}\sum_i A^{\mathrm{DAER}}(\tau_i) &= \frac{\sum_i (\tilde{A}^{\mathrm{DAER}}(\tau_i) - \mu_{\mathrm{DAER}})}{N(\sigma_{\mathrm{DAER}} + \epsilon_{\mathrm{norm}})} \\ &= 0. \end{aligned} \tag{33}$$

Dividing by $\sigma_{\mathrm{DAER}} + \epsilon_{\mathrm{norm}}$ gives the variance in Eq. (31).

For ranking, if $A^E(\tau_i) = A^E(\tau_j)$, then

$$\begin{aligned} &\tilde{A}^{\mathrm{DAER}}(\tau_i) - \tilde{A}^{\mathrm{DAER}}(\tau_j) \\ &\quad = \eta p\big(\bar{\Delta} t(\tau_j) - \bar{\Delta} t(\tau_i)\big). \end{aligned} \tag{34}$$

This quantity is positive when $\bar{\Delta} t(\tau_i) < \bar{\Delta} t(\tau_j)$ and $\eta p > 0$. The normalization in Eq. (29) subtracts the same mean and divides by the same positive value, so the ordering is preserved. □

A.5 Combining Trajectory-Level and Step-Level Advantages

AEPO combines trajectory-level and step-level guidance through

$$A^{\mathrm{AEPO}}_{i,t} = A^{\mathrm{DAER}}_i + \omega A^{\mathrm{ACSD}}_{i,t}. \tag{35}$$

DAER adjusts the trajectory-level reward advantage using arrival delay relative to successful rollouts. ACSD computes step-level advantages using a baseline weighted by preceding-action similarity. The coefficient ω controls the relative contribution of ACSD to the combined advantage, which is used in the clipped policy objective.

B Complete Experimental Details

B.1 Training Configuration

We follow the training protocol of GiGPO, using matched settings for methods evaluated on the same benchmark.

Model Configuration. We use Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct for ALFWorld and WebShop, and Qwen2.5-3B-Instruct for Search-R1.

Table 2: Training configurations and common hyperparameter settings for ALFWorld and WebShop.

Task-Specific Configuration			Common Hyperparameters	
Setting	**ALFWorld**	**WebShop**	**Parameter**	**Value**
Maximum episode length	50 steps	30 steps	Step discount (γ)	0.95
Maximum prompt length	2048 tokens	4096 tokens	Balance (ω)	1.0
Training iterations	150	150	Group size (N)	8
			Learning rate	10^{-6}
			Clip ratio (ϵ_{clip})	0.2

Hyperparameter Settings. For ALFWorld and WebShop, the shared settings are a discount factor of $\gamma = 0.95$, balance coefficient $\omega = 1.0$, group size $N = 8$, actor learning rate 10^{-6}, and clip ratio $\epsilon_{\text{clip}} = 0.2$. The normalization stabilizer is $\epsilon_{\text{norm}} = 10^{-6}$, distinct from the clipping threshold ϵ_{clip}. The ACSD temperature is fixed at $\tau = 0.1$ except in the temperature sensitivity experiment.

Task-Specific Settings. The RL results in Table 1 use 3 random seeds, with the mean and standard deviation reported across seeds. All methods use identical configurations for fair comparison.

ALFWorld and WebShop. Table 2 summarizes the task-specific budgets and common hyperparameters. Each rollout for group-based methods samples 16 task groups with eight trajectories per group, yielding 128 environment instances; PPO uses 128 independently initialized instances. Responses are capped at 512 tokens. Sampling temperatures are 1.0 during training and 0.4 during validation. PPO uses a critic learning rate of 10^{-5}. Successful and failed episodes receive rewards of 10 and 0, respectively, with a penalty of -0.1 for invalid actions. The optimization mini-batch sizes are 256 for ALFWorld and 64 for WebShop, and the KL loss coefficient is 0.01 for both benchmarks.

Search-Augmented Question Answering. Search-R1 uses a maximum prompt length of 4096 tokens, a maximum response length of 512 tokens, and up to four interaction turns. Each training batch contains 256 questions, with five rollouts sampled per question. The actor learning rate is 10^{-6}. Training uses a mini-batch size of 512 and a KL loss coefficient of 0.001. Sampling temperatures are 1.0 for training and 0.0 for validation. The reward is 1 for success and 0 for failure, with an invalid-action penalty of -0.01. The discount factor and step-level advantage weight are fixed at $\gamma = 0.95$ and $\omega = 1$.

Training Budget. ALFWorld and WebShop experiments run for 150 training iterations, and Search-R1 experiments run for 200 iterations.

B.2 Benchmark Descriptions

ALFWorld. An embodied environment designed to assess multi-step decision-making abilities of LLM agents. In each episode, the agent receives a text-based goal (e.g., "put a hot apple in the fridge") and must accomplish it through multi-turn interaction with the environment. The benchmark contains 3,827 task instances across six categories of common household activities: Pick & Place (Pick), Examine in Light (Look), Clean & Place (Clean), Heat & Place (Heat), Cool & Place (Cool), and Pick Two & Place (Pick2).

WebShop. A complex web-based interactive environment designed to test LLM agents in realistic online shopping scenarios. To complete each task, the agent must interact with a simulated HTML-based shopping website to search for, navigate to, and ultimately purchase a suitable item matching user requirements (e.g., "buy cheap wireless headphones with good reviews").

Table 3: Task categories in the ALFWorld benchmark.

Category	Description
Pick & Place (Pick)	Locate and move objects to target locations.
Examine in Light (Look)	Pick up objects and examine them under lamps.
Clean & Place (Clean)	Clean objects and place them appropriately.
Heat & Place (Heat)	Heat objects in microwaves before placing them.
Cool & Place (Cool)	Cool objects in fridges before placing them.
Pick Two & Place (Pick2)	Manipulate two objects sequentially before placing them.

Table 4: Environment specifications of the WebShop benchmark.

Component	Details
Product catalog	Over 1.1 million real products collected from Amazon.
User instructions	12,000 diverse shopping instructions across product categories.
Observations	HTML-based observations requiring semi-structured parsing.
Action space	Search queries, page navigation, and attribute-based filtering.
Evaluation metrics	Score for attribute matching and Success for task completion.
Episode constraint	A maximum of 30 interaction steps per episode.

C Additional Comparisons with Recent RL Methods

Table 5 compares RAGEN (StarPO-S) (Wang et al., 2025) and ProxMO (Fang et al., 2026) with GiGPO and AEPO using Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct. The comparisons use matched training and evaluation settings. AEPO achieves the highest overall ALFWorld and WebShop success rates among these methods at both model scales.

Table 5: Additional baseline comparisons on ALFWorld and WebShop. ALFWorld entries are task success rates (%); WebShop reports score and success rate (%). **Bold**: best result within each model scale.

Method	ALFWorld							WebShop	
	Pick	Look	Clean	Heat	Cool	Pick2	All	Score	Succ.
Qwen2.5-1.5B-Instruct									
RAGEN (StarPO-S)	87.45	75.93	85.18	84.94	77.31	72.52	80.45	80.54	63.84
GiGPO	93.94	81.35	96.70	95.24	81.82	82.87	89.06	83.18	68.75
ProxMO	93.66	**89.28**	89.38	92.20	**86.52**	86.60	90.24	83.69	68.91
AEPO (Ours)	**94.90**	87.04	**100.00**	**100.00**	80.93	**95.26**	**92.97**	**84.05**	**71.88**
Qwen2.5-7B-Instruct									
RAGEN (StarPO-S)	93.00	84.92	91.70	90.02	81.28	82.87	87.66	83.41	66.94
GiGPO	96.29	84.07	93.75	90.11	87.53	88.89	92.19	83.35	75.78
ProxMO	**97.56**	86.85	93.78	90.71	**89.03**	88.98	92.49	85.92	76.77
AEPO (Ours)	97.30	**100.00**	**97.62**	**100.00**	85.30	**94.74**	**95.31**	**90.21**	**80.47**

D MULTI-TURN RETRIEVAL WITH FREE-FORM QUERIES

We evaluate AEPO with Qwen2.5-3B-Instruct in the Search-R1 setting (Jin et al., 2025) on HotpotQA, 2Wiki, MuSiQue, and Bamboogle. The agent generates free-form search queries and uses retrieved passages to answer questions through multiple interaction turns. Following the GiGPO setup, observations are grouped by similarity. We compare Search-R1 (GRPO), GiGPO, and AEPO under the same training and evaluation settings.

As shown in Table 6, AEPO achieves the highest score on all four datasets. Compared with GiGPO, it improves scores by 3.27 points on HotpotQA, 2.14 on 2Wiki, 0.74 on MuSiQue, and 3.20 on Bamboogle.

Table 6: Results on four question-answering datasets in the Search-R1 setting using Qwen2.5-3B-Instruct. **Bold** indicates the best score on each dataset.

Method	HotpotQA	2Wiki	MuSiQue	Bamboogle
Search-R1 (GRPO)	30.70	29.07	9.80	29.77
GiGPO	33.90	32.33	12.33	29.87
AEPO	**37.17**	**34.47**	**13.07**	**33.07**

E ACTION ENCODER COMPARISON

We compare TF-IDF with two dense action encoders, all-MiniLM-L6-v2 and bge-large-en-v1.5, by replacing the action representation in ACSD while keeping the remaining training and evaluation settings fixed.

MiniLM achieves success rates close to those of TF-IDF on both benchmarks. BGE obtains the highest success rates, improving over TF-IDF by 0.53 percentage points on ALFWorld and 0.68 on WebShop. Its advantage-computation overhead reaches 312.58% and 260.42%, respectively, compared with 5.60% and 4.11% for TF-IDF. These results support TF-IDF as an efficient choice for action representation in ACSD.

Table 7: Comparison of action encoders in ACSD. Overhead denotes the percentage increase in advantage-computation time relative to GiGPO. **Bold** indicates the highest success rate or lowest overhead.

Action encoder	ALFWorld		WebShop	
	Success (%) ↑	Adv. overhead (%) ↓	Success (%) ↑	Adv. overhead (%) ↓
TF-IDF	92.97	**5.60**	71.88	**4.11**
all-MiniLM-L6-v2	92.94	37.32	72.14	25.80
bge-large-en-v1.5	**93.50**	312.58	**72.56**	260.42

F Difficulty Scaling and Group Normalization

We examine the contributions of difficulty scaling and group normalization in DAER. The variant without difficulty scaling removes the group success-rate factor from the delay penalty while retaining group normalization. The variant without both components additionally removes mean centering and standard-deviation scaling of the delay-adjusted trajectory advantages.

Removing difficulty scaling reduces success rates from 92.97% to 89.53% on ALFWorld and from 71.88% to 69.81% on WebShop. Removing both components further reduces success rates to 85.24% and 67.43%, respectively. The full model also achieves the highest success rates on the Look, Heat, and Pick2 task categories.

Table 8: Ablation of difficulty scaling and group normalization in DAER. Success rates are reported as percentages. Step counts are averaged over all evaluation trajectories. **Bold** indicates the highest success rate in each column.

Method	ALFWorld					WebShop	
	Look	Heat	Pick2	All	Steps	Succ.	Steps
GiGPO	81.35	95.24	82.87	89.06	24.22	68.75	14.54
AEPO w/o difficulty scaling	82.03	92.20	85.81	89.53	18.26	69.81	11.24
AEPO w/o scaling & normalization	76.58	87.27	80.92	85.24	14.58	67.43	9.88
AEPO (full)	**87.04**	**100.00**	**95.26**	**92.97**	21.91	**71.88**	13.02

Historical repetition across task difficulty. We stratify rollout groups by their success rate p and compare trajectory success rates between rollouts with fewer than three hard-stuck steps and those with at least three. As shown in Table 9, more hard-stuck steps are associated with lower success rates in the three higher-success-rate buckets. In the lowest-success-rate bucket, success rates are 5.9% for rollouts with fewer hard-stuck steps and 6.8% for those with more. This variation motivates scaling the DAER penalty by p, which reduces efficiency pressure when successful rollouts are scarce.

Table 9: Trajectory success rates across group-success-rate buckets, stratified by the number of hard-stuck steps. Success rates are reported as percentages; the gap is the rate for at least three hard-stuck steps minus that for fewer than three, in percentage points.

Group success rate p	Trajectory success rate (%)		Gap (pp)
	Hard-stuck steps < 3	Hard-stuck steps ≥ 3	
$[0, 0.25)$	5.9	6.8	$+0.9$
$[0.25, 0.5)$	37.1	31.6	-5.5
$[0.5, 0.75)$	64.4	51.5	-12.9
$[0.75, 1]$	95.9	87.3	-8.6

G Worked Examples

We present two trajectory excerpts drawn from AEPO (Qwen2.5-7B-Instruct) rollouts to concretely illustrate the two structural bottlenecks corrected by our framework and how the ACSD and DAER components address them.

G.1 Preceding-Action Bias: Same Observation, Different Outcomes

The following two trajectory windows both encounter the identical degenerate observation "`Nothing happens.`" in ALFWorld bathroom-cleaning tasks. They differ **only** in the preceding causal intent (action).

Table 10: **Successful continuation.** The agent proceeds to task success after observing `Nothing happens.`

Step	Observation	Action
7	`You move the soapbar 1 to the countertop 1.`	`examine soapbar 1`
8	`Nothing happens.`	`examine countertop 1`
9	**`On the countertop 1, you see a... soapbar 1...`**	`take soapbar 1 from countertop 1`

Table 11: **Failure caused by repeated unproductive actions.** After reaching a `Nothing happens.` state, the agent repeatedly selects `go to sinkbasin 1`, remains stuck, and eventually fails at the step limit.

t	Observation	Action / Outcome
20	`Nothing happens.`	`go to sinkbasin 1` *(stuck)*
21	`Nothing happens.`	`go to sinkbasin 1` *(stuck)*
22	`Nothing happens.`	*repeats until failure*

Trajectory A — `examine soapbar 1` → `Nothing happens.` → **Task Success** (score = 10)
Task: `put a clean soapbar in countertop`.

Agent reasoning at step 9: "The task is to put a clean soapbar... I have already examined the soapbar 1 and confirmed that it is clean. The next logical step is to pick it up..." The "`Nothing happens.`" at step 8 is a non-destructive `examine` result—the agent correctly interprets it as confirmation that the soapbar is clean and proceeds to task completion.

Trajectory B — `go to sinkbasin 1` → `Nothing happens.` → **Task Failure** (score = 0)
Task: `put a clean soapbar in countertop` (same task, different rollout).

Credit Assignment Comparison. Across 82,000 rollout trajectories, the empirical success rates conditioned on the preceding action at "`Nothing happens.`" steps diverge drastically: `examine soapbar 1` yields 0.72, while `go to sinkbasin 1` yields 0.14.

GiGPO pools all steps sharing this textual observation, computing a heavily skewed uniform baseline (e.g., 0.43) that erroneously penalizes the productive `examine` path and improperly rewards the aimless navigation path. In contrast, AEPO's **Action-Conditioned Spatial Disambiguation (ACSD)** mechanism evaluates the TF-IDF cosine similarity of these preceding actions. The preceding actions `examine soapbar 1` and `go to sinkbasin 1` exhibit near-zero lexical overlap. At a sufficiently low temperature, ACSD assigns a soft weight $w_{ij} \approx 0$ between them. This isolates the spatial noise, naturally synthesizing accurate, action-specific baselines ($\hat{\mu}_\tau^{(i)} \approx 0.72$ and 0.14, respectively) without requiring sparse hard-clustering.

G.2 Temporal Waste: The Echo Trap and Recovery

The following is a complete 6-step successful trajectory (AEPO 7B) for the task "`put some toiletpaper on garbagecan`". The optimal path requires 4 steps; this agent wastes 2 steps on a malformed, repeated action before recovering.

- **Step 1 & 2:** Agent attempts to reach the toilet paper but outputs a malformed navigation action (`toiletpaperhanger 1` instead of `go to...`). The environment responds with `Nothing happens.`. The agent repeats the exact same error (*hard-stuck echo*).
- **Step 3 & 4 (Recovery):** Agent corrects its behavior, navigates to `countertop 1`, and successfully picks up `toiletpaper 1`.
- **Step 5 & 6:** Agent navigates to `garbagecan 1` and places the object. Task completes (score = 10).

Table 12: **Temporal delay in an inefficient trajectory.** The first two steps do not advance task completion, causing subsequent useful states to lag behind their earliest attainable steps.

t	**State** s_t	$t_{\min}$	Δt
1	`Nothing happens.`[†]	1	0
2	`Nothing happens.`[‡]	1	1
3	arrive at `countertop 1`	1	2
4	pick up `toiletpaper 1`	2	2
5	arrive at `garbagecan 1`	3	2
6	place `toiletpaper 1`	4	2

[†]Malformed action; [‡]repeated failed action.

Temporal Regularization Analysis. For this calculation, consider a reference group containing the successful 4-step and 6-step trajectories, and map the first two `Nothing happens.` observations to the same state key. The 4-step path reaches the useful states at steps 1, 2, 3, and 4. The successful 6-step path itself establishes $t_{\min} = 1$ for the repeated observation; its first two delays are therefore 0 and 1. The remaining four arrivals are each delayed by two steps:

The mean delay is $\bar{\Delta t}(\tau) = \frac{0+1+2+2+2+2}{6} = 1.50$, compared with zero for the 4-step path. At $\eta = 0.03$, the uncentered delay term is $\eta p \bar{\Delta t} = 0.045p$; DAER subsequently centers and normalizes the adjusted advantages, so this is not the final normalized advantage difference. Under the stated invalid-action penalty, the two failed actions also reduce the 6-step return to $10 - 2 \times 0.1 = 9.8$, versus 10 for the 4-step path. DAER provides an explicit delay-sensitive signal in addition to this reward difference.

H Related Work

H.1 LLMs as Decision-Making Agents

Recent literature has witnessed a paradigm shift in deploying Large Language Models (LLMs) as autonomous agents across interactive environments, ranging from embodied tasks (Shridhar et al., 2020) and web navigation (Furuta et al., 2024; Zhang et al., 2024) to strategic gameplay (Wang et al., 2023). Initial methodologies heavily relied on frozen pre-trained models, utilizing sophisticated prompting mechanisms like ReAct (Yao et al., 2022b) and Reflexion (Shinn et al., 2023), or integrating external memory and tools (Schick et al., 2023). While these heuristic approaches established fundamental baselines, their reliance on static knowledge severely limits their robustness against out-of-distribution environmental dynamics. Consequently, the research frontier has advanced toward parameter-updating techniques. By employing Supervised Fine-Tuning (SFT) (Zhang & Zhang, 2024) and Reinforcement Learning (RL) (Sutton et al., 1998), current methods aim to cultivate adaptive, end-to-end decision-making capabilities directly from multi-turn interaction trajectories.

H.2 Group-Based RL for LLMs

Applying RL to align and optimize LLMs was originally dominated by actor-critic architectures (e.g., PPO (Schulman et al., 2017)). However, the necessity of maintaining distinct value networks introduces prohibitive memory and computational overhead during large-scale model training. To circumvent this, group-relative optimization paradigms—such as RLOO (Kool et al.; Ahmadian et al., 2024), GRPO (Shao et al., 2024), and DAPO (Yu et al., 2026)—have gained significant traction. By estimating policy advantages through the empirical variance of multiple responses generated from a single prompt, these critic-free methods deliver exceptional efficiency. Yet, their success is largely confined to single-turn domains with immediate algorithmic verification (e.g., code generation or mathematical reasoning) (Guo et al., 2025). When transplanted to multi-turn agentic environments, relying on sparse, episode-level rewards results in overly coarse credit assignment, severely hindering long-horizon convergence.

H.3 Stepwise Credit Assignment in Long-Horizon Tasks

To address the inefficiency of trajectory-level rewards, recent advancements attempt to achieve fine-grained, stepwise credit assignment. Approaches like RAGEN (Wang et al., 2025) evaluate trajectories by concatenating entire historical contexts, which inadvertently triggers context length explosion and limits scalability. Alternatively, stepwise methods deconstruct trajectories into isolated decision points. GiGPO (Feng et al., 2026) pioneers this by clustering steps that share identically matched observations to compute local advantages. However, this strict matching protocol yields sparse clusters in high-dimensional spaces and collapses under observation degeneracy. Concurrent works attempt to mitigate these sampling inefficiencies: STEP (Chen et al., 2025) dynamically allocates rollouts based on task difficulty, while ProxMO (Fang et al., 2026) softens discrete state boundaries through continuous proximity weighting.

Step-level methods such as GiGPO evaluate actions using discounted future returns, without explicitly measuring the historical redundancy accumulated along a trajectory. Furthermore, isolating environmental observations from the agent's preceding causal intent leads to profound context inconsistency. In this work, **AEPO** directly confronts these core bottlenecks. AEPO combines action-conditioned step-level baselines with a success-rate-weighted trajectory delay penalty.