TSPO: Breaking the Double Homogenization Dilemma in Multi-turn Search Policy Optimization

Shichao Ma[1,2*] **Zhiyuan Ma**[2] **Ming Yang**[3,2*] **Xiaofan Li**[4,2*] **Xing Wu**[2]
Jintao Du[2] **Yu Cheng**[2] **Weiqiang Wang**[2] **Qiliang Liu**[5]
Zhengyang Zhou[1†] **Yang Wang**[1†]

[1]University of Science and Technology of China [2]Tiansuan Lab, Ant Group Co., Ltd.
[3]Fudan University [4]East China Normal University [5]Central South University

Abstract

Multi-turn tool-integrated reasoning enables Large Language Models (LLMs) to solve complex tasks through iterative information retrieval. However, current reinforcement learning (RL) frameworks for search-augmented reasoning predominantly rely on sparse outcome-level rewards, leading to a "Double Homogenization Dilemma." This manifests as (1) *Process homogenization*, where the thinking, reasoning, and tooling involved in generation are ignored. (2) *Intra-group homogenization*, coarse-grained outcome rewards often lead to inefficiencies in intra-group advantage estimation with methods like Group Relative Policy Optimization (GRPO) during sampling. To address this, we propose **T**urn-level **S**tage-aware **P**olicy **O**ptimization (**TSPO**). TSPO introduces the First-Occurrence Latent Reward (FOLR) mechanism, allocating partial rewards to the step where the ground-truth answer first appears, thereby preserving process-level signals and increasing reward variance within groups without requiring external reward models or any annotations. Extensive experiments demonstrate that TSPO significantly outperforms state-of-the-art baselines, achieving average performance gains of **24%** and **13.6%** on Qwen2.5-3B and 7B models, respectively. Code is available at `https://github.com/Flipped-May/TSPO`.

1 Introduction

Large Language Models (LLMs) have recently shown strong abilities in complex reasoning through multi-turn interactions with external tools such as search engines, calculators, and code interpreters (Team et al., 2025; Jin et al., 2025; Chai et al., 2025). Unlike single-step decision-making, this multi-turn paradigm allows models to decompose complex tasks into a series of manageable sub-tasks. By iteratively retrieving and integrating information, LLMs have achieved remarkable success in challenging tasks like open-domain question answering (QA) and mathematical reasoning (Zhang et al., 2025b; DESIGN, 2025).

Despite their potential, current reinforcement learning (RL) frameworks for multi-turn tool calling predominantly rely on outcome-level reward signals (Jin et al., 2025; Chen et al., 2025). In tasks like QA, performance is typically measured by Exact Match (EM) (Maalouly, 2022), where a binary reward is assigned only at the final turn. While this avoids potential *reward hacking* (Amodei et al., 2016), ***it compresses the entire multi-turn dynamic tooling and reasoning process into a single scalar***, obfuscating the quality of intermediate steps. We identify that this reliance on sparse, outcome rewards creates a systemic bottleneck, which we term the **Double Homogenization Dilemma**.

The first facet is **process-level reward homogenization**, where trajectories with diverse intermediate reasoning or retrieval quality receive identical outcome rewards. This phenomenon ignores progress made before the final step and incorrectly penalizes beneficial turn-level actions, such as successful information acquisition. The second facet is **intra-group reward homogenization**, which is particularly problematic in Group Relative Policy Optimization (GRPO) (Shao et al., 2024). Since the binary nature of rewards often results in uniform-reward trajectory groups, it eliminates variance and causes vanishing advantages. This ultimately nullifies policy gradients and discards potentially valuable reasoning trajectories.

Existing research attempts to mitigate these issues via process-level supervision, such as LLM-based reasoning scoring (Zhang et al., 2025c; Yuan et al., 2025) or search-based strategies like MCTS (Zhang et al., 2025a; Wang et al., 2025b). Although providing finer-grained feedback, these methods incur heavy overheads from expensive annotations,

*Work done during internship at Ant Group.
†Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2026, pages 28607–28623

reliance on proprietary models, and often suffer from limited generalizability across tasks and tools.

To address this dilemma without extra overhead, we propose **T**urn-level **S**tage-aware **P**olicy **O**ptimization (**TSPO**). TSPO adopts the First-Occurrence Latent Reward (**FOLR**) mechanism, which detects the earliest turn containing the ground-truth. For the first facet, TSPO grants partial reward to such turns even if the final answer is wrong, retaining useful process signals. For the second facet, this turn-level allocation increases variance within trajectory groups, avoiding vanishing advantages in uniform-reward groups. Crucially, TSPO achieves this without requiring any external reward models, or additional human labels.

Contributions. Contributions are as follows:

- Our analysis and experiments confirm the "Double Homogenization Dilemma" (i.e., process and intra-group homogenization) in multi-turn RL and its negative impact.
- We propose **TSPO**, which first locates the earliest occurrence of the ground-truth via the **FOLR** mechanism, then allocates partial rewards to preserve process signals, and finally performs turn-level advantage estimation, without requiring any external judges or human annotation overhead.
- Extensive experiments across 7 diverse QA datasets demonstrate that TSPO significantly outperforms state-of-the-art baselines, achieving substantial accuracy gains of **24%** and **13.6%** on Qwen2.5-3B and 7B models, respectively.

2 Double Homogenization Dilemma

The *Double Homogenization Dilemma* manifests through two coupled facets: process-level and intra-group homogenization. In this section, we analyze each phenomenon and their negative impact.

2.1 Process-level Reward Homogenization

Multi-turn QA task typically proceeds in two coupled stages: (1) iterative query reformulation and external information retrieval; and (2) final answer synthesis based on the accumulated evidence. Formally, given an input query $x \sim \mathcal{D}$, the agent generates a trajectory:

$$y = \left[l_1, f_1, \ldots, \underbrace{l_i, f_i}_{i\text{-th turn}}, \ldots, l_k, f_k, a \right] \quad (1)$$

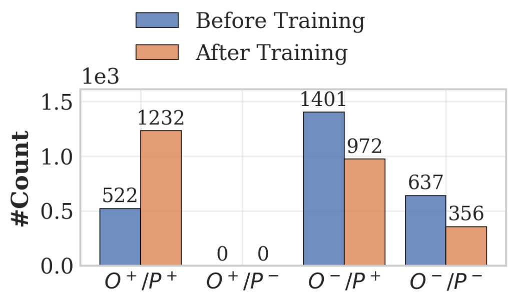

Figure 1: Reasoning trajectory distribution (O vs. P) for Qwen2.5-7B-Instruct: no retrieval-free successes (O^+/P^-); near-miss (O^-/P^+) and total-failure cases (O^-/P^-) treated equally.

where l_i and f_i represent the reasoning thoughts and environment feedback at turn i, and a is the final model answer. Let a_{gold} be the ground-truth answer for x. Outcome-level RL assigns a sparse outcome reward:

$$r_{\text{outcome}}(x, y) = \mathbb{I}[a = a_{\text{gold}}] \in \{0, 1\}, \quad (2)$$

which *compresses the entire multi-turn reasoning process into a single binary scalar*.

Empirical analysis reveals that a_{gold} often appears in retrieved feedback f_i at some intermediate turn, yet the model fails to synthesize it correctly, yielding an incorrect final answer (details see Appendix F). Under outcome-level supervision, both *retrieval failures* ($a_{\text{gold}} \notin \{f_i\}_{i=1}^k$) and *synthesis failures* ($a_{\text{gold}} \in \{f_i\}_{i=1}^k$ but $a \neq a_{\text{gold}}$) receive identical zero rewards, discarding partial successes in information acquisition.

To quantify this issue, we evaluate trajectories along two orthogonal dimensions: **Outcome Accuracy** (O): whether $a = a_{\text{gold}}$; and **Process Integrity** (P): whether a_{gold} appears in any retrieved feedback $\{f_i\}_{i=1}^k$. This yields four categories:

- O^-/P^-: Complete failure where the model neither retrieves nor synthesizes the correct answer.
- O^-/P^+: Near-Miss where correct evidence retrieved but synthesis failed.
- O^+/P^+: Full success where both retrieval and synthesis stages are correct.
- O^+/P^-: Retrieval-Free Success where the final answer is correct without ever retrieving a_{gold}.

Figure 1 compares the prevalence of these categories across normal training stages. Two key observations emerge: (1) Retrieval-Free Success (O^+/P^-) remains absent, indicating that successful retrieval is a prerequisite for correct synthesis. (2) Within failed attempts (O^-), both Near-Miss (O^-/P^+) and Total Failure (O^-/P^-) occur but receive the same zero reward.

This inability to distinguish retrieval success from failure exemplifies ***process-level reward homogenization***, where the convergence of heterogeneous intermediate reasoning qualities results in indistinguishable reward signals, erasing fine-grained progress cues and hindering the optimization of tool calling and reasoning.

2.2 Intra-Group Reward Homogenization

Group Relative Policy Optimization (GRPO) (Shao et al., 2024) stabilizes training by normalizing rewards within groups of trajectories sampled from the same question x. For a group of size G, let $\{y_i\}_{i=1}^{G}$ be the sampled trajectories, each assigned a binary outcome reward $r_{\text{outcome},i}(x, y_i) = \mathbb{I}[a_i = a_{\text{gold}}] \in \{0, 1\}$. The per-trajectory advantage is computed as:

$$\hat{A}_i = \frac{r_i - \mu_r}{\sigma_r + \epsilon}, \quad (3)$$

where μ_r and σ_r are the mean and standard deviation of rewards in the group, and ϵ is a small constant for numerical stability. The policy gradient is proportional to $\hat{A}_i$.

When $\sigma_r = 0$ (i.e., all trajectories in the group receive identical rewards), advantages vanish ($\hat{A}_i \equiv 0$), and the group contributes no gradient updates. Because ***process-level reward homogenization*** collapses many Near-Miss (O^-/P^+) trajectories into the same 0 reward as complete failures, the probability of sampling *uniform-reward groups* (i.e., $\forall i, j \in \{1, \ldots, G\},\ r_i = r_j \wedge i \neq j, r_i \in \{1, 0\}$) is greatly increased.

To quantify this, we empirically quantify the composition of training groups ($G = 5$ rollouts) by counting the proportion of all-correct, all-wrong, and mixed groups across training steps' rollouts (Figure 2). Two observations emerge:

- **High frequency of all-wrong groups**: due to binary rewards, the occurrence of uniform-reward groups is significant, with all-wrong groups maintaining a consistently high proportion ($\geq 40\%$) throughout the normal training.
- **Stalled optimization from all-wrong groups**: As training progresses, the increase in all-correct groups mainly comes from mixed groups that provide non-zero variance, whereas the proportion of all-wrong groups barely changes.

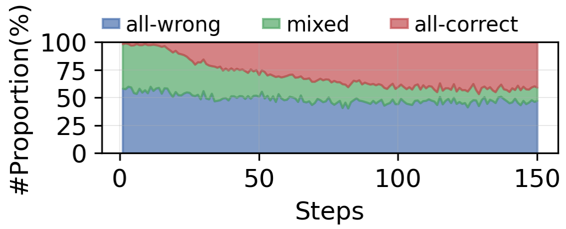

Figure 2: Trajectory groups with rollout size $G = 5$, composed by `all-correct`, `mixed` and `all-wrong`, during training on Qwen2.5-7B-Instruct.

We term this phenomenon ***intra-group reward homogenization***: the disappearance of discriminative signals within sampling groups, which prevents the optimizer from exploiting many trajectories and constitutes the second stage of the Double Homogenization Dilemma.

3 Turn-level Stage-aware Policy Optimization (TSPO)

To enable fine-grained reward assignment without external supervision, we propose **Turn-level Stage-aware Policy Optimization (TSPO)**. Unlike prior methods that rely on costly step-level annotations or external reward models (Zhang et al., 2025c; Yuan et al., 2025; Zhang et al., 2025a; Wang et al., 2025b), TSPO operates solely on the standard input–answer pairs (x, a_{gold}) and derives turn-level rewards directly from the model's retrieving/reasoning process, thereby addressing the Double Homogenization Dilemma while avoiding additional computation overhead.

3.1 The First-Occurrence Hypothesis: Identifying Latent Signals

First-Occurrence Hypothesis The presence of a_{gold} in the intermediate retrieved feedback f_i, can serve as a latent signal of partial progress. Formally, for any retrieval feedback $f_i \in y$, if $a_{\text{gold}} \in f_i$, then the signal strongly foreshadows the success of the final answer a.

Hypothesis Test To validate the hypothesis, we analyze the relationship between *Evidence Presence (P)* and *Outcome Accuracy (O)* as defined in Section 2.1 on 51,713 test trajectories. We build a 2×2 table (Table 1) that cross-tabulates outcomes along these two binary dimensions. To test whether

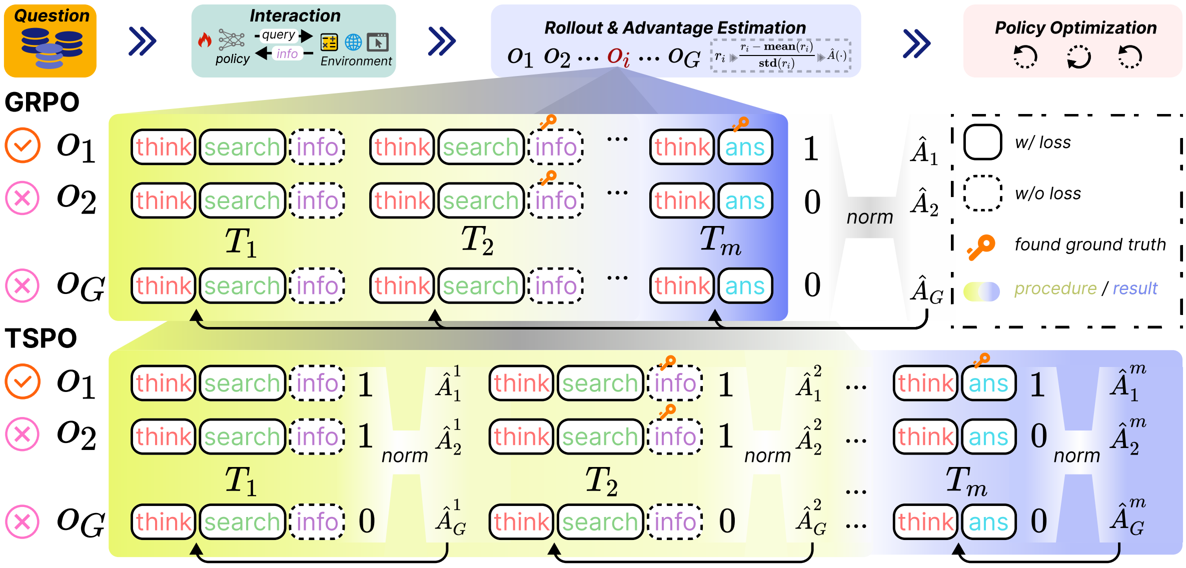

Figure 3: **Overview of TSPO.** Unlike outcome-level RL which assigns identical zero rewards to O_2 (O^-/P^+, middle) and O_G (O^-/P^-, bottom), TSPO identifies the first occurrence (t^*, red key) of a_{gold} to assign turn-level rewards and performs advantage estimation on a per-turn basis. This restores intra-group variance and provides fine-grained signals. A detailed walkthrough of this example is provided in Section 3.2.

O and P are statistically independent, we perform Pearson's chi-squared test (Plackett, 1983).

Table 1: Correlation analysis between outcome accuracy (O) and evidence presence (P) for Qwen2.5-7B-Instruct

	O^+	O^-	Total Num
P^+	10,092	25,645	35,737
P^-	0	15,976	15,976
Total Num	10,092	41,621	51,713

Under the null hypothesis of independence, the expected frequency for cell (i, j) is:

$$E_{i,j} = \frac{(\text{number of } i\text{-th row}) \times (\text{number of } j\text{-th column})}{\text{total example number}} \tag{4}$$

The chi-squared test statistic is then computed as:

$$\chi^2 = \sum_{i,j \in \{1,2\}} \frac{(O_{i,j} - E_{i,j})^2}{E_{i,j}} \tag{5}$$

where $O_{i,j}$ denotes the observed count in cell (i, j) from Table 1. Substituting the observed and expected frequencies yields $\chi^2 = 5605.5$, which corresponds to a p-value less than 0.001, indicating a highly significant dependency between evidence presence and final correctness.

Notably, all O^+ cases (10,092/10,092) also satisfy P^+, while P^- cases *never* achieve O^+, confirming that retrieval of a_{gold} is a *necessary condition* for success. Hence, we incorporate this validated signal, the first-occurrence of the ground-truth into TSPO's reward allocation.

3.2 The TSPO Framework: Reward Assignment and Optimization

Building on the First-Occurrence hypothesis, TSPO reformulates sparse outcome rewards into dense, turn-level signals for policy optimization.

Turn-level MDP Formulation. We integrate the validated First-Occurrence signal into a turn-level Markov Decision Process (MDP) (Puterman, 1990), i.e., $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$, where the state $s_k \in \mathcal{S}$ encodes the dialogue history and retrieved evidence up to turn k, the action $a_k \in \mathcal{A}$ is either a query to an external tool or a synthesis step, $t(s_{k+1} \mid s_k, a_k) \in \mathcal{T}$ is the environment transition updating the reasoning state given a_k and feedback f_k, and $r(s_k, a_k) \in \mathcal{R}$ is the turn-level reward defined by the FOLR mechanism.

Reward Assignment with FOLR. Formally, the *First-Occurrence Latent Reward* (FOLR) assigns turn-level reward based on the earliest occurrence of the ground-truth answer in the reasoning trace.

Specifically, given the first-occurrence turn t^* where a_{gold} appears in trajectory i, the reward $r_{i,k}$ for each turn k is defined as:

$$r_{i,k} = \begin{cases} 1 & \text{if } a_i = a_{\text{gold}} \\ \alpha & \text{if } a_i \neq a_{\text{gold}} \text{ and } k \leq t^* \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $\alpha \in [0, 1]$ is a partial reward coefficient.

Turn-level Advantage Estimation. TSPO replaces the trajectory-level scalar r_i in vanilla GRPO with per-turn rewards $r_{i,k}$. The advantage for trajectory i at turn k is computed via group-relative normalization:

$$\hat{A}_{i,k} = \frac{r_{i,k} - \mathrm{mean}(r_{\cdot,k})}{\mathrm{std}(r_{\cdot,k}) + \epsilon}, \tag{7}$$

where $r_{i,k}$ denotes the set of rewards for all trajectories in the sampling group that contain turn k. This turn-level design ensures that Near-Miss (O_2) trajectories remain distinguishable from Total Failures (O_G), restoring the discriminative gradient signals within the group.

Policy Optimization with TSPO. Replacing GRPO's binary r_i and trajectory-level advantage with the FOLR reward in Eq (6) and per-turn advantage Eq (7), the optimization objective becomes:

$$\mathcal{J}_{\mathrm{TSPO}}(\theta) = \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G}\sum_{j=1}^{k}\mathcal{L}_{i,j} - \beta D_{\mathrm{KL}}(\pi_\theta \,\|\, \pi_{\mathrm{ref}})\right]. \tag{8}$$

where $x \sim \mathcal{D}, y_i \sim \pi_\theta$ and the per-turn loss $\mathcal{L}_{i,j}$ is given by

$$\mathcal{L}_{i,j} = \sum \min\left(w_{i,t}\hat{A}_{i,j},\ \mathrm{clip}\left(w_{i,t},\ 1 \pm \epsilon\right)\hat{A}_{i,j}\right). \tag{9}$$

$w_{i,t} = \frac{\pi_\theta(a_{i,t}|s_{i,t})}{\pi_{\mathrm{old}}(a_{i,t}|s_{i,t})}$ is the importance coefficient for action $a_{i,t}$ at t-th token in turn k of trajectory i, and $\hat{A}_{i,k}$ is the normalized per-turn advantage from Eq (7). By computing advantages and updating policies at the turn level, TSPO preserves informative process signals for *Near-Miss* (O^-/P^+) trajectories and introduces reward variance within groups, mitigating both process-level and intra-group reward homogenization.

Example: Turn-level Reward and Advantage To illustrate how TSPO assigns rewards and computes advantages, consider a sampling group of $G = 3$ trajectories for the same question, as depicted in Figure 3.

As established in Section 2.1, the case O^+/P^- (producing the correct answer without retrieving the ground-truth) is empirically absent. We therefore concentrate on three observed trajectory types:

- **Trajectory O_1 (O^+/P^+):** The model retrieves the ground-truth a_{gold} at turn t^* and correctly synthesizes it in the final answer ($a = a_{\mathrm{gold}}$).
- **Trajectory O_2 (O^-/P^+):** The model retrieves a_{gold} at turn t^* but fails to produce the correct final answer ($a \neq a_{\mathrm{gold}}$).
- **Trajectory O_G (O^-/P^-):** The model never retrieves a_{gold} and produces an incorrect answer.

Table 2: Turn-level rewards assigned by TSPO ($\alpha = 1$) for three trajectory types.

Trajectory	Turn 1	Turn 2	Turn 3	Turn 4
O_1 (O^+/P^+)	1	1	1	1
O_2 (O^-/P^+)	1	1	0	0
O_G (O^-/P^-)	0	0	0	0

Table 2 lists the turn-level rewards assigned by TSPO under partial reward coefficient $\alpha = 1$. Here, "1" denotes full or partial reward based on Equation (6), while "0" signifies a zero reward value assigned to that turn.

Turn-level advantages are computed via group normalization. For Turn 1 (rewards = [1, 1, 0]), mean $\mu \approx 0.67$ and $\sigma \approx 0.47$, yielding: $\hat{A}_{O_{1,1}} = \hat{A}_{O_{2,1}} \approx +0.71$, $\hat{A}_{O_{G,1}} \approx -1.41$. For Turn 3 (rewards = [1, 0, 0]), $\hat{A}_{O_{1,3}} \approx +1.41$, $\hat{A}_{O_{2,3}} = \hat{A}_{O_{G,3}} \approx -0.71$.

This shows that TSPO distinguishes Near-Miss (O_2) from total failure (O_G) by assigning positive rewards up to t^*, restoring intra-group variance even in all-wrong groups, while GRPO gives zero advantage to both.

Discussion 1: Variable-length Trajectories. For groups with variable numbers of turns, we pad shorter trajectories to the group's maximum turn using their final-turn reward. This allows $\mathrm{mean}(r_{\cdot,k})$ and $\mathrm{std}(r_{\cdot,k})$ to be computed over all G trajectories at each turn, ensuring consistent variance. Padded turns are masked during optimization and thus do not affect gradient updates.

Discussion 2: Group Types. Groups can be classified as all-correct, all-wrong, and mixed. All-correct groups (O_1 only) retain trajectory-level allocation as in GRPO. All-wrong groups (O_2 and O_G only) are the main target of TSPO; per-turn allocation ensures O_2 trajectories contribute gradients. Mixed groups naturally contain reward variance, and no special handling is applied. The focus on all-wrong groups is motivated by their high prevalence, as shown in Section 2.2, and will be further analyzed in our ablation Section 5.1.

Table 3: Main results. The best performance is set in bold. †/⋆ represents in-domain/out-domain datasets.

Methods	General QA			Multi-Hop QA				
	NQ†	TriviaQA⋆	PopQA⋆	HotpotQA†	2wiki⋆	Musique⋆	Bamboogle⋆	Avg.
Qwen2.5-7b-Instruct								
Direct Inference	0.134	0.408	0.140	0.183	0.250	0.031	0.120	0.181
IRCoT	0.224	0.478	0.301	0.133	0.149	0.072	0.224	0.239
Search-o1	0.151	0.443	0.131	0.187	0.176	0.058	0.296	0.206
RAG	0.349	0.585	0.392	0.299	0.235	0.058	0.208	0.304
SFT	0.318	0.354	0.121	0.217	0.259	0.066	0.112	0.207
R1-instruct	0.270	0.537	0.199	0.237	0.292	0.072	0.293	0.271
Rejection Sampling	0.360	0.592	0.380	0.331	0.296	0.123	0.355	0.348
Search-R1	0.393	0.610	0.397	0.377	0.404	0.146	0.368	0.385
ZeroSearch	0.436	0.652	**0.488**	0.346	0.352	0.184	0.278	0.391
MT-PPO	0.498	0.649	0.459	0.428	0.404	0.215	-	-
StepSearch	-	-	-	0.386	0.366	**0.226**	0.400	-
ReasonRAG	-	-	0.415	0.384	**0.436**	0.128	0.360	-
TSPO	**0.527**	**0.687**	**0.488**	**0.430**	0.407	0.151	**0.416**	**0.444**
Qwen2.5-3b-Instruct								
Direct Inference	0.106	0.288	0.108	0.149	0.244	0.020	0.024	0.134
IRCoT	0.111	0.312	0.200	0.164	0.171	0.067	0.240	0.181
Search-o1	0.238	0.472	0.262	0.221	0.218	0.054	0.320	0.255
RAG	0.348	0.544	0.387	0.255	0.226	0.047	0.080	0.270
SFT	0.249	0.292	0.104	0.186	0.248	0.044	0.112	0.176
R1-instruct	0.210	0.449	0.171	0.208	0.275	0.060	0.192	0.224
Rejection Sampling	0.294	0.488	0.332	0.240	0.233	0.059	0.210	0.265
Search-R1	0.341	0.545	0.378	0.324	0.319	0.103	0.264	0.325
ZeroSearch	0.414	0.574	0.448	0.274	0.300	0.098	0.111	0.317
StepSearch	-	-	-	0.345	0.328	**0.174**	**0.344**	-
ReasonRAG	-	-	0.329	0.300	0.266	0.069	0.136	-
TSPO	**0.490**	**0.655**	**0.472**	**0.384**	**0.364**	0.126	0.328	**0.403**

4 Experiments

4.1 Experimental Setup

Datasets We evaluate TSPO on seven benchmark datasets, categorized as follows: (1) **General Question Answering**: NQ (Kwiatkowski et al., 2019), TriviaQA (Joshi et al., 2017), and PopQA (Mallen et al., 2022). (2) **Multi-Hop Question Answering**: HotpotQA (Yang et al., 2018), 2WikiMultiHopQA (Ho et al., 2020), Musique (Trivedi et al., 2022b), and Bamboogle (Press et al., 2022). These datasets provide a diverse range of search with reasoning challenges, enabling a comprehensive evaluation.

Metric We report Exact Match score (Maalouly, 2022), which is the standard metric for QA.

Training Details We use Qwen2.5-3B-Instruct and Qwen2.5-7B-Instruct (Team et al., 2024) as the base models, E5 (Wang et al., 2022) for retrieval, and the 2018 Wikipedia dump (Karpukhin et al., 2020) as the corpus.

Baselines To evaluate the effectiveness of TSPO, we compare it against the following baselines: (1) **Inference without Retrieval**: Direct inference and Chain-of-Thought (CoT) reasoning (Wei et al., 2022). (2) **Inference with Retrieval**: Retrieval-Augmented Generation (RAG) (Lewis et al., 2020), IRCoT (Trivedi et al., 2022a), and Search-o1 (Li et al., 2025). (3) **Fine-Tuning-Based Methods**: Supervised fine-tuning (SFT) (Chung et al., 2024), RL-based fine-tuning without a search engine (R1) (Guo et al., 2025) and rejection sampling (Ahn et al., 2024) with a search engine. (4) **Agentic RL**: Search-R1 (Jin et al., 2025), ZeroSearch (Sun et al., 2025), MT-PPO (DESIGN, 2025), StepSearch (Wang et al., 2025b) and ReasonRAG (Zhang et al., 2025a).

4.2 Main Results

Overall comparison. Table 3 summarizes results on both general QA and multi-hop QA benchmarks. Across all datasets, TSPO achieves the highest average performance on both Qwen2.5-7B-Instruct and Qwen2.5-3B-Instruct. Compared to the strongest baseline, TSPO delivers relative improvements of **+13.6%** and **+24.0%** for the 7B and 3B models respectively. The consistent gains across in-domain (†) and out-of-domain (⋆) datasets verify the robustness and general applicability of our method.

Compared with sparse outcome-level rewards. Relative to outcome-level RL methods such as Search-R1 and ZeroSearch, TSPO attains superior accuracy. This benefit stems from alleviating the *Double Homogenization Dilemma*, enabling the model to leverage partial-success trajectories that

Table 4: **Ablation study on the Group Type.** Adapt all-wrong groups yields better performance.

Method	Group Type	HotpotQA	2Wiki	MuSiQue	Bamboogle	Avg.
TSPO	*None*	0.370	0.338	0.100	0.208	0.254
TSPO	*All groups*	**0.388**	0.363	0.125	0.304	0.295
TSPO	*All-wrong groups*	0.384	**0.364**	**0.126**	**0.328**	**0.300**

would otherwise yield zero rewards. By converting these trajectories into informative gradient signals, TSPO facilitates more effective policy updates during training.

Compared with dense intermediate rewards. Dense-reward methods such as StepSearch, MT-PPO, and ReasonRAG introduce more fine-grained reward assignment than sparse outcome-level RL. While these signals improve learning granularity, they typically require more complex infrastructure, curated resources, or heavy computation. As shown in Table 3, TSPO consistently outperforms all three methods despite using only (x, a_{gold}) supervision. This demonstrates that the *First-Occurrence Latent Reward* (FOLR) signal, although simple to obtain, is highly informative and sufficient to drive effective reward assignment.

5 Further Study

5.1 Ablation Study on the Group Types

We conduct an ablation study to examine the impact of applying TSPO's turn-level advantage allocation to different *Group Types* within GRPO. Recall from Section 2.2 that groups can be classified as *all-correct*, *all-wrong*, or *mixed*. We compare three configurations:

- **None**: TSPO reverts to GRPO's trajectory-level advantage allocation, providing no mitigation for process or intra-group reward homogenization.
- **All groups**: turn-level reward assignment is applied to all categories of types.
- **All-wrong groups**: turn-level reward assignment is restricted to all-wrong groups, while all-correct and mixed groups retain trajectory-level rewards.

Analysis. The results are presented in Table 4. Compared to *None*, both *All groups* and *All-wrong group* achieve substantial gains. The drop in *None* is expected: maintaining GRPO's trajectory-level advantage fails to address either process-level or intra-group reward homogenization, leaving partial-success trajectories unused.

Interestingly, *All-wrong group* slightly outperforms *All groups*. We attribute this to three factors: (1) All-correct groups already represent fully successful trajectories and do not benefit from per-turn reward assignment. (2) Mixed groups contain intrinsic reward variance, which GRPO can already exploit without modification. (3) All-wrong groups are both high in prevalence and homogeneous in reward under GRPO, making them the primary source of wasted samples. Thus, selectively enhancing signal in all-wrong groups is sufficient and slightly more effective to unlock the full benefit of FOLR, without unnecessary computation on already-informative groups.

5.2 Analysis of Training Dynamics

To examine how TSPO mitigates the Double Homogenization Dilemma at the optimization level, we track the training dynamics of the policy model across three metrics: policy entropy, KL divergence, and gradient norm. As shown by the logs in Figure 4, we compare the baseline (GRPO) with two TSPO variants: *TSPO+All groups* and *TSPO+All-wrong groups*.

Policy Diversity and Entropy Collapse. In the baseline, policy entropy decays rapidly to near zero, indicating *policy collapse*: the model overfits to a few trajectories that happen to pass final verification, suppressing exploration of alternative reasoning paths. This behavior stems from the sparsity and binarity of outcome-level rewards. In contrast, both TSPO variants maintain a substantially higher entropy, showing that turn-level stage-aware signals preserve diversity in tool-use and reasoning by attaching value to informative intermediate steps.

Stability and KL Divergence. KL divergence measures deviation from the base model's distribution. The baseline reaches a higher KL value (∼0.02) with zero entropy, implying drift toward a deterministic yet potentially suboptimal policy. TSPO variants stably converge around 0.01, benefiting from FOLR-based partial rewards that refine search quality without discarding linguistic and rea-

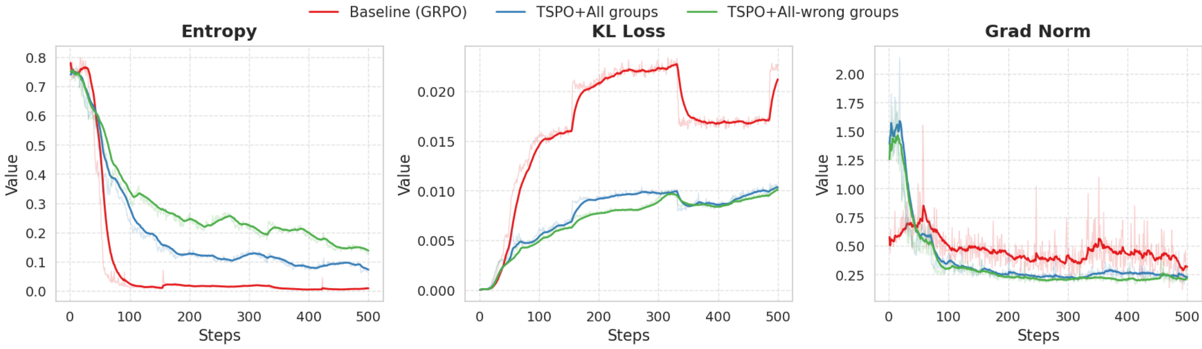

Figure 4: Comparison of Training Dynamics. **Left**: Policy entropy collapses to near zero in GRPO but remains stable under TSPO variants, preserving reasoning diversity. **Middle**: KL divergence from GRPO spikes in the baseline, indicating unstable policy drift; TSPO maintains consistent alignment. **Right**: Gradient norms are large and volatile in GRPO due to sparse rewards, while TSPO yields smoother, more consistent updates.

soning priors. This stabilizes optimization while avoiding sharp shifts away from the base model.

Optimization Consistency via Gradient Norms. Gradient norm reflects the magnitude and steadiness of policy updates. The baseline's higher value ($\sim$0.5) and greater variance align with the intuition that sparse rewards yield vanishing advantages for most samples but cause abrupt updates for the few winners. TSPO reduces and smooths updates ($\sim$0.25) by supplying dense turn-level reward, shaping a steadier optimization landscape.

Table 5: **The training time.** We analyze the per-step training time of different methods.

Method	Group Type	Time Cost per Step
Qwen2.5-3B-Instruct		
Search-R1	-	243.9s
TSPO	*All groups*	278.7s
TSPO	*All-wrong groups*	268.6s
Qwen2.5-7B-Instruct		
Search-R1	-	414.7s
TSPO	*All groups*	446.0s
TSPO	*All-wrong groups*	402.3s

5.3 Training Cost

We compare the per-step training time of TSPO with Search-R1 under two group allocation strategies: *All groups* and *All-wrong groups* (Table 5). Experiments are conducted with a batch size of 512 on 8 NVIDIA A100 GPUs, with additional training details provided in Appendix B.

Overall, despite introducing turn-level advantage allocation, the training time of TSPO is comparable to that of Search-R1. This is because TSPO does not rely on any external judge or reward model; the proposed FOLR mechanism derives reward allocation directly from (x, a_{gold}) pairs available within the model, without incurring additional inference or annotation cost. The higher time for *All groups* compared to *All-wrong groups* is due to the increased number of finer-grained turn-level groups that must be processed during advantage calculation. This leads to additional computation in the grouping stage, slightly raising the training cost.

Notably, when using Qwen2.5-7B-Instruct, TSPO on *All-wrong groups* achieves lower per-step time than Search-R1. We attribute this to the fact that Search-R1 assigns identical rewards to all failed trajectories, failing to distinguish their quality. This lack of discriminative feedback hinders learning and results in redundant and repetitive search actions during rollouts, thereby increasing overall training time.

6 Conclusion

We identify and formalize the *Double Homogenization Dilemma* in multi-turn tool-augmented reasoning: (1) *process homogenization*, where outcome-only rewards collapse behaviorally distinct trajectories into identical scalar signals; and (2) *intra-group homogenization*, where uniform-reward groups in GRPO yield vanishing gradients. To address both issues, we propose **TSPO**, a turn-level policy optimization framework centered on the **First-Occurrence Latent Reward** (FOLR) mechanism. FOLR allocates partial reward to the earliest turn containing the ground truth, requiring only (x, a_{gold}) supervision. Experiments across seven QA benchmarks demonstrate that TSPO consistently outperforms strong baselines, while maintaining computational efficiency.

7 Acknowledgements

This paper is partially supported by the National Natural Science Foundation of China (No.62502488, No.12227901), Natural Science Foundation of Jiangsu Province (BK20240460).

8 Limitations

TSPO shows strong empirical performance and favorable computational properties, but still has limitations:

- **Dependence on retrieval correctness.** FOLR assumes that the correct answer must appear in retrieved evidence. Tasks solvable without direct retrieval may require modified reward definitions.
- **Domain extensibility.** Our experiments focus on **search-augmented reasoning**. The applicability of TSPO and the Double Homogenization theory to other domains (e.g., code generation, multi-step tool use) remains open.

Future work will explore adapting TSPO to broader task types, updating FOLR for settings where retrieval is imperfect or absent.

References

Janice Ahn, Rishu Verma, Renze Lou, Di Liu, Rui Zhang, and Wenpeng Yin. 2024. Large language models for mathematical reasoning: Progresses and challenges. *arXiv preprint arXiv:2402.00157*.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. 2016. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*.

Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. 2012. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43.

Huacan Chai, Zijie Cao, Maolin Ran, Yingxuan Yang, Jianghao Lin, Xin Peng, Hairui Wang, Renjie Ding, Ziyu Wan, Muning Wen, and 1 others. 2025. Parl-mt: Learning to call functions in multi-turn conversation with progress awareness. *arXiv preprint arXiv:2509.23206*.

Mingyang Chen, Linzhuang Sun, Tianpeng Li, Haoze Sun, Yijie Zhou, Chenzheng Zhu, Haofen Wang, Jeff Z Pan, Wen Zhang, Huajun Chen, and 1 others. 2025. Learning to reason with search for llms via reinforcement learning. *arXiv preprint arXiv:2503.19470*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, and 1 others. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

AGENTS VIA TURN-LEVEL REWARD DESIGN. 2025. Reinforcing multi-turn reasoning in llm agents via turn-level reward design.

Guanting Dong, Hangyu Mao, Kai Ma, Licheng Bao, Yifei Chen, Zhongyuan Wang, Zhongxia Chen, Jiazhen Du, Huiyang Wang, Fuzheng Zhang, and 1 others. 2025. Agentic reinforced policy optimization. *arXiv preprint arXiv:2507.19849*.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. 2025. Group-in-group policy optimization for llm agent training. *arXiv preprint arXiv:2505.10978*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Zhezheng Hao, Hong Wang, Haoyang Liu, Jian Luo, Jiarui Yu, Hande Dong, Qiang Lin, Can Wang, and Jiawei Chen. 2025. Rethinking entropy interventions in rlvr: An entropy change perspective. *arXiv preprint arXiv:2510.10150*.

Zhezheng Hao, Hong Wang, Jian Luo, Jianqing Zhang, Yuyan Zhou, Qiang Lin, Can Wang, Hande Dong, and Jiawei Chen. 2026. Recreate: Reasoning and creating domain agents driven by experience. *arXiv preprint arXiv:2601.11100*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. *arXiv preprint arXiv:2011.01060*.

Jinwei Hu, Yi Dong, Youcheng Sun, and Xiaowei Huang. 2026a. Tapas are free! training-free adaptation of programmatic agents via llm-guided program synthesis in dynamic environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 29477–29485.

Jinwei Hu, Xinmiao Huang, Youcheng Sun, Yi Dong, and Xiaowei Huang. 2026b. Lying with truths: Open-channel multi-agent collusion for belief manipulation via generative montage. *arXiv preprint arXiv:2601.01685*.

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

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pages 6769–6781.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, and 1 others. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. 2025. Search-o1: Agentic search-enhanced large reasoning models. *arXiv preprint arXiv:2501.05366*.

Shiyu Liu, Yongjing Yin, Jianhao Yan, Yunbo Tang, Qinggang Zhang, Bei Li, Xin Chen, Jingang Wang, Xunliang Cai, and Jinsong Su. 2026. Bapo: Boundary-aware policy optimization for reliable agentic search. *arXiv preprint arXiv:2601.11037*.

Yunbo Long, Yunhan Liu, and Liming Xu. 2026. Emomas: Emotion-aware multi-agent system for high-stakes edge-deployable negotiation with bayesian orchestration. *Preprint*, arXiv:2604.07003.

Shichao Ma, Yunhe Guo, Jiahao Su, Qihe Huang, Zhengyang Zhou, and Yang Wang. 2026. Talk2image: A multi-agent system for multi-turn image generation and editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 32437–32445.

Shichao Ma, Xinfeng Zhang, Zeng Zhao, Bai Liu, Changjie Fan, and Zhipeng Hu. 2025. Dialogdraw: Image generation and editing system based on multi-turn dialogue. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 24795–24803.

Nicolas El Maalouly. 2022. Exact matching: Algorithms and related problems. *arXiv preprint arXiv:2203.13899*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. 2022. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*, 7.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Robin L Plackett. 1983. Karl pearson and the chi-squared test. *International statistical review/revue internationale de statistique*, pages 59–72.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2022. Measuring and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*.

Martin L Puterman. 1990. Markov decision processes. *Handbooks in operations research and management science*, 2:331–434.

Cheng Qian, Emre Can Acikgoz, Qi He, Hongru Wang, Xiusi Chen, Dilek Hakkani-Tür, Gokhan Tur, and Heng Ji. 2025. Toolrl: Reward is all tool learning needs. *arXiv preprint arXiv:2504.13958*.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

Hao Sun, Zile Qiao, Jiayan Guo, Xuanbo Fan, Yingyan Hou, Yong Jiang, Pengjun Xie, Yan Zhang, Fei Huang, and Jingren Zhou. 2025. Zerosearch: Incentivize the search capability of llms without searching. *arXiv preprint arXiv:2505.04588*.

Rui Sun, Jie Ding, Chenghua Gong, Tianjun Gu, Yihang Jiang, Juyuan Zhang, Liming Pan, and Linyuan Lü. 2026. Topodim: One-shot topology generation of diverse interaction modes for multi-agent systems. *arXiv preprint arXiv:2601.10120*.

Qwen Team and 1 others. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2(3).

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, and 1 others. 2025. Tongyi deepresearch technical report. *arXiv preprint arXiv:2510.24701*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022a. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. *arXiv preprint arXiv:2212.10509*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022b. Musique: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Hanlin Wang, Chak Tou Leong, Jiashuo Wang, Jian Wang, and Wenjie Li. 2025a. Spa-rl: Reinforcing llm agents via stepwise progress attribution. *arXiv preprint arXiv:2505.20732*.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Ziliang Wang, Xuhui Zheng, Kang An, Cijun Ouyang, Jialu Cai, Yuhang Wang, and Yichao Wu. 2025b. Stepsearch: Igniting llms search ability via step-wise proximal policy optimization. *arXiv preprint arXiv:2505.15107*.

Ziqing Wang, Yibo Wen, William Pattie, Xiao Luo, Weimin Wu, Jerry Yao-Chieh Hu, Abhishek Pandey, Han Liu, and Kaize Ding. 2025c. Polo: Preference-guided multi-turn reinforcement learning for lead optimization. *arXiv preprint arXiv:2509.21737*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Fei Wu, Zhenrong Zhang, Qikai Chang, Jianshu Zhang, Quan Liu, and Jun Du. 2026. Step potential advantage estimation: Harnessing intermediate confidence and correctness for efficient mathematical reasoning. *arXiv preprint arXiv:2601.03823*.

Guangzhi Xiong, Qiao Jin, Xiao Wang, Yin Fang, Haolin Liu, Yifan Yang, Fangyuan Chen, Zhixing Song, Dengyu Wang, Minjia Zhang, and 1 others. 2025. Rag-gym: Optimizing reasoning and search agents with process supervision. *arXiv preprint arXiv:2502.13957*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.

Jiahao Yuan, Zhiqing Cui, Hanqing Wang, Yuansheng Gao, Yucheng Zhou, and Usman Naseem. 2025. Kardia-r1: Unleashing llms to reason toward understanding and empathy for emotional support via rubric-as-judge reinforcement learning. *arXiv preprint arXiv:2512.01282*.

Wenlin Zhang, Xiangyang Li, Kuicai Dong, Yichao Wang, Pengyue Jia, Xiaopeng Li, Yingyi Zhang, Derong Xu, Zhaocheng Du, Huifeng Guo, and 1 others. 2025a. Process vs. outcome reward: Which is better for agentic rag reinforcement learning. *arXiv preprint arXiv:2505.14069*.

Wenyuan Zhang, Xinghua Zhang, Haiyang Yu, Shuaiyi Nie, Bingli Wu, Juwei Yue, Tingwen Liu, and Yongbin Li. 2026a. Expseek: Self-triggered experience seeking for web agents. *Preprint*, arXiv:2601.08605.

Yaocheng Zhang, Haohuan Huang, Zijun Song, Yuanheng Zhu, Qichao Zhang, Zijie Zhao, and Dongbin Zhao. 2025b. Criticsearch: Fine-grained credit assignment for search agents via a retrospective critic. *arXiv preprint arXiv:2511.12159*.

Yuzhe Zhang, Feiran Liu, Yi Shan, Xinyi Huang, Xin Yang, Yueqi Zhu, Xuxin Cheng, Cao Liu, Ke Zeng, Terry Jingchen Zhang, and 1 others. 2026b. Silobench: A scalable environment for evaluating distributed coordination in multi-agent llm systems. *arXiv preprint arXiv:2603.01045*.

Yuzhe Zhang, Xianwei Xue, Xingyong Wu, Mengke Chen, Chen Liu, Xinran He, Run Shao, Feiran Liu, Huanmin Xu, Qiutong Pan, and 1 others. 2026c. Don't act blindly: Robust gui automation via action-effect verification and self-correction. *arXiv preprint arXiv:2604.05477*.

Zhenru Zhang, Chujie Zheng, Yangzhen Wu, Beichen Zhang, Runji Lin, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. 2025c. The lessons of developing process reward models in mathematical reasoning. *arXiv preprint arXiv:2501.07301*.

Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. 2025. Deepresearcher: Scaling deep research via reinforcement learning in real-world environments. *arXiv preprint arXiv:2504.03160*.

Keyang Zhong, Junlin Xie, Hefeng Wu, Haofeng Li, and Guanbin Li. 2026. Collaborative multi-agent scripts generation for enhancing imperfect-information reasoning in murder mystery games. *Preprint*, arXiv:2604.11741.

Yifei Zhou, Song Jiang, Yuandong Tian, Jason Weston, Sergey Levine, Sainbayar Sukhbaatar, and Xian Li. 2025. Sweet-rl: Training multi-turn llm agents on collaborative reasoning tasks. *arXiv preprint arXiv:2503.15478*.

Wei Zhu, Zhiwen Tang, and Kun Yue. 2026. Symphony: Synergistic multi-agent planning with heterogeneous language model assembly. *arXiv preprint arXiv:2601.22623*.

A Related Work

Tool-integrated Reasoning and Search. The integration of external tools, particularly search engines, has significantly enhanced the factuality and multi-hop reasoning capabilities of large language models (LLMs) (Schick et al., 2023; Ma et al., 2025; Zhu et al., 2026). Early retrieval-augmented generation (RAG) methods focused on single-turn retrieval to alleviate hallucinations (Lewis et al., 2020; Xiong et al., 2025). Recent advancements have shifted toward agentic and iterative search, where models such as IRCoT (Trivedi et al., 2022a) and Search-o1 (Li et al., 2025) reformulate queries and reason across multiple turns (Wang et al., 2025c). ReAct (Yao et al., 2022) demonstrates another effective direction by using prompting to guide iterative reasoning together with search engine calls, enabling models to interleave reasoning steps and tool usage within a single trajectory. While these approaches improve information acquisition, most rely on fixed heuristics or supervised fine-tuning (SFT) (Ouyang et al., 2022), which may not fully exploit the model's exploratory potential in complex environments.

Reinforcement Learning for Reasoning. Reinforcement Learning (RL) has emerged as a powerful paradigm for optimizing reasoning paths (Hao et al., 2025; Liu et al., 2026; Zhang et al., 2026c). Models like DeepSeek-R1 (Guo et al., 2025) and Search-R1 (Jin et al., 2025) utilize outcome-level rewards to reinforce successful trajectories. Specifically, the Group Relative Policy Optimization (GRPO) (Shao et al., 2024) algorithm has gained prominence for its stability and efficiency in group-based normalization. However, relying solely on outcome rewards leads to sparse supervision in long-horizon tasks. Recent work such as DeepResearcher (Zheng et al., 2025; Dong et al., 2025) integrates search agents with real-world web interactions, enriching training signals with dynamic and diverse environments. In addition, other studies (Hao et al., 2026; Sun et al., 2026; Ma et al., 2026; Hu et al., 2026b) explore multi-agent collaboration, where different specialized agents work (Zhong et al., 2026; Long et al., 2026; Hu et al., 2026a; Zhang et al., 2026b) together to achieve better reasoning and performance. Our work builds upon the GRPO framework but identifies a critical bottleneck, namely the homogenization of rewards within sampling groups, which hampers the optimization of near-successful attempts.

Process Rewards Assignment. To address reward sparsity, researchers have explored process-level supervision (Zhou et al., 2025; Zhang et al., 2026a; Wu et al., 2026). Process Reward Models (PRMs) (Zhang et al., 2025c) and step-wise verifiers provide feedback for individual reasoning steps, although they typically require expensive human annotations or calls to superior teacher models such as GPT-4. Other approaches, such as ReasonRAG (Zhang et al., 2025a) and StepSearch (Wang et al., 2025b), utilize Monte Carlo Tree Search (MCTS) (Browne et al., 2012) or information-gain metrics to generate dense rewards. Some models (Qian et al., 2025; Yang et al., 2025) design explicit process reward mechanisms to evaluate intermediate reasoning steps and tool usage, yet these designs are often tailored to specific tools or datasets, which limits their general applicability. While effective, many of these methods incur significant computational overhead or depend on predefined gold evidence (Wang et al., 2025a; Feng et al., 2025). TSPO diverges from these works by extracting a latent "first-occurrence" signal directly from the reasoning trace, enabling fine-grained reward assignment without external judges or additional simulation costs, and providing a lightweight solution for stage-aware policy optimization.

B Experimental Setups

Our implementation is based on *Search-R1* (Jin et al., 2025), and all training procedures are conducted using the *Verl* framework (Sheng et al., 2024). We perform experiments on two model scales: Qwen-2.5-3B and Qwen-2.5-7B (Team et al., 2024). We train each model for a total of 500 steps. For GRPO training, the group size (G) is set to 5. The entropy coefficient is fixed at 0, and the KL-loss coefficient is set to 1×10^{-3}. Training is performed on 8 NVIDIA A100 GPUs. The total batch size is 512, with a mini-batch size of 256 and a micro-batch size of 8 per GPU.

C Resource Details

Table 6 summarizes all resources used in our experiments, including datasets, models, and reference codebases. For each dataset, we list the number of examples and provide the corresponding URL for public access. The seven QA datasets cover both single-hop and multi-hop scenarios, as well as in-domain and out-of-domain settings, enabling a diverse evaluation of search-augmented reasoning. The model entries include the Qwen2.5-3B-Instruct and Qwen2.5-7B-Instruct variants, which serve as our base policy models during reinforcement learning. Reference libraries contain open-source implementations used in training, including Verl for RLHF infrastructure and Search-R1 as the baseline framework for our experiments. All listed resources are publicly available, ensuring that our setup can be reproduced without reliance on proprietary data or closed-source software.

D Train Details

D.1 Training Prompt

Prompt 1: Training Prompt

Answer the given question. You must conduct reasoning inside <think> and </think> first every time you get new information. After reasoning, if you find you lack some knowledge, you can call a search engine by <tool_call> {"name": <function-name>, "arguments": <args-json-object>} </tool_call> and it will return the top searched results between <tool_response> and </tool_response>. You can search as many times as your want. If you find no further external knowledge needed, you can directly provide the answer inside <answer> and </answer>, without detailed illustrations. For example, <answer> Beijing </answer>. Question:

D.2 Train Rewards

Figure 5 shows the training reward curves of three methods: GRPO (outcome-level baseline), TSPO with turn-level rewards applied only to *all-wrong groups*, and TSPO applied to *all groups* across four multi-hop QA datasets: HotpotQA, 2WikiMulti-HopQA, MuSiQue, and Bamboogle.

We observe that GRPO exhibits slow and unstable reward growth across all datasets, often plateauing early. This reflects the Double Homogenization Dilemma: sparse binary rewards cause most trajectory groups (especially all-wrong ones) to yield zero advantages, severely limiting learning signal. In contrast, both TSPO variants achieve consistently faster and smoother reward convergence. Notably, even though TSPO+All-wrong modifies rewards only in homogeneous failure groups, it matches or nearly matches the performance of TSPO+All groups, which confirmes our analysis in Section 5.1 that all-wrong groups are the primary bottleneck in standard GRPO training.

The accelerated reward learning under TSPO demonstrates that leveraging first-occurrence latent signals effectively transforms previously wasted near-miss trajectories into useful supervision, enabling more sample-efficient policy optimization.

D.3 Reasoning Efficiency and Response Length

We monitor the average response length (in tokens) throughout training. All models begin with verbose reasoning trajectories (over 2,100 tokens) but gradually converge to more concise outputs (1,700–1,900 tokens). Crucially, while outcome-based RL baselines often resort to aggressive shortening that sacrifices necessary retrieval or reasoning steps, TSPO variants achieve compression through improved reasoning efficiency. This indicates that turn-level supervision guides the policy to eliminate redundancy without discarding critical intermediate actions.

D.4 Training Algorithm of TSPO

We present the complete training procedure of *Turn-level Stage-aware Policy Optimization* (TSPO), which integrates the First-Occurrence Latent Reward (FOLR) mechanism into a group-based reinforcement learning framework. TSPO supports two practical advantage allocation strategies: `All groups`, which applies turn-level normalization to all trajectory groups, and `All-wrong groups`, which applies it only to homogeneous failure groups. This selective design is motivated by our analysis in Section 2.2, which shows that all-wrong groups are the primary source of wasted samples under standard GRPO.

As outlined in Algorithm 1, TSPO proceeds in three stages per training batch. First, it performs multi-turn rollouts to collect diverse reasoning trajectories for each question. Second, it assigns turn-level rewards using FOLR: unsuccessful trajectories receive partial reward (α) for all turns up to and including the first appearance of a_{gold}, while successful trajectories retain full outcome reward across all turns. Third, it computes advantages either at the turn level or the trajectory level, depending on the group composition and the chosen strategy. This enables TSPO to preserve fine-grained signals from near-miss attempts and mitigate both

Table 6: Overview of datasets, models, and reference libraries used in our experiments, including sizes and URLs.

Type	Size/Params	URL
Dataset		
Natural Questions (NQ) (Kwiatkowski et al., 2019)	100,231	https://ai.google.com/research/NaturalQuestions
TriviaQA (Joshi et al., 2017)	847,579	https://huggingface.co/datasets/mandarjoshi/trivia_qa
PopQA (Mallen et al., 2022)	14,267	https://huggingface.co/datasets/akariasai/PopQA
HotpotQA (Yang et al., 2018)	203,109	https://hotpotqa.github.io/
2WikiMultiHopQA (Ho et al., 2020)	192,606	https://github.com/Alab-NII/2wikimultihop
Musique (Trivedi et al., 2022b)	22,355	https://github.com/stonybrooknlp/musique
Bamboogle (Press et al., 2022)	125	https://huggingface.co/datasets/cmriat/bamboogle
Model		
Qwen2.5-3B-Instruct (Team et al., 2024)	3B	https://huggingface.co/Qwen/Qwen2.5-3B-Instruct
Qwen2.5-7B-Instruct (Team et al., 2024)	7B	https://huggingface.co/Qwen/Qwen2.5-7B-Instruct
Reference library		
Verl (Sheng et al., 2024)	-	https://github.com/volcengine/verl
Search-R1 (Jin et al., 2025)	-	https://github.com/PeterGriffinJin/Search-R1

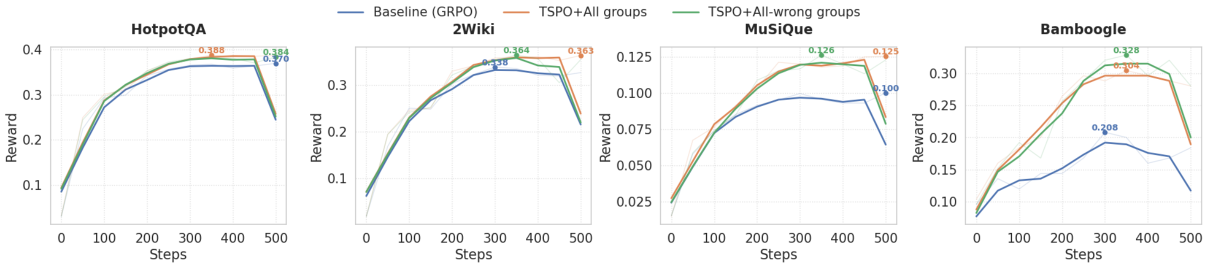

Figure 5: Training reward curves on four representative datasets. TSPO variants show consistently faster and more stable reward convergence compared to the GRPO baseline.

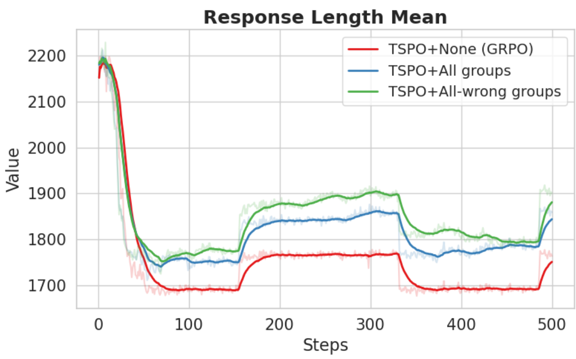

Figure 6: **Response length evolution during training.** TSPO variants reduce trajectory length steadily while preserving reasoning integrity, avoiding the premature truncation observed in outcome-only methods.

process homogenization and intra-group homogenization.

E Hyperparameter Analysis

We evaluate the influence of the partial reward coefficient α in the FOLR mechanism (Section 3.2), considering $\alpha \in \{0, 0.5, 1\}$. The results are presented in Figure 7.

When $\alpha = 0$, TSPO degenerates into standard GRPO since all incorrect trajectories are assigned zero rewards across all turns. This leads to intra-group homogenization in all-wrong groups, where Near-Miss trajectories (O_2) and complete failures (O_G) become indistinguishable in terms of reward signals. Such homogenization erases potential process-level cues, causes vanishing advantages, and ultimately yields the lowest performance.

For $\alpha = 0.5$ and $\alpha = 1$, intra-group variance is restored by assigning positive rewards to turns preceding the first occurrence of a_{gold} in Near-Miss (O_2) trajectories. The average performances are similar because group-wise normalization (Equation 7) preserves the relative ordering of trajectories, where O_2 consistently obtains a normalized advantage around $+1$ whereas O_G obtains approximately -1. This indicates that any $\alpha > 0$ is sufficient to mitigate intra-group reward homogenization, with the precise value having minimal effect on final performance.

F Case Study: Partial Successes Penalized as Total Failures

Under outcome-level supervision, any deviation from the exact ground-truth answer results in a zero

Algorithm 1 TSPO Training Stage

Require: Policy π_θ, Reference π_{ref}, Dataset $\mathcal{D}$, Ground-truth a_{gold}, Group size G, Reward coeff. α, Strategy $\in \{\texttt{All-groups}, \texttt{All-wrong}\}$
Ensure: Optimized policy π_θ

```
Rollout: Sample batch D_b ⊂ D
for q ∈ D_b do
    Sample G trajectories {y_i}_{i=1}^G ~ π_θ(q)
    FOLR:
    for y_i do
        r_i^out ← 𝕀(EM(y_i, a_gold))
        t* ← first turn with a_gold ∈ Response_{i,t}
        for t = 1 to T_i do
            r_{i,t} ← { α        if t ≤ t* ∧ r_i^out = 0
                      { r_i^out  otherwise
        end for
    end for
    Advantage Estimation:
    r̄ ← (1/G) Σ r_i^out
    if Strategy = All-wrong and r̄ = 0 then
        for t do
            A_{i,t} ← (r_{i,t} − mean({r_{j,t}})) / (std({r_{j,t}}) + ε)
        end for
    else if Strategy = All-groups then
        Compute turn-level A_{i,t} for all groups
    else
        A_i ← (r_i^out − mean(r^out)) / (std(r^out) + ε)
    end if
end for
Policy Update: Optimize π_θ w.r.t. advantages {A_{i,t}}
return π_θ
```

reward, regardless of whether critical evidence was successfully retrieved. This collapses trajectories with meaningful partial progress into the same failure category as those with no progress at all.

We present two such failure cases from our training logs (Table 7). In the first, the model is asked for the tissue type of the innermost layer of cells. The correct answer is *Epithelium*. After one search turn, multiple documents (Docs 1, 2, 4) explicitly state that "Epithelium is one of the four basic types of animal tissue", directly providing a_{gold}. However, the model misinterprets biological context and outputs *Endoderm*, confusing germ layers with tissue types.

In the second case, the model searches for the first Nobel Prize in Physics winner. The retrieved results include both "Wilhelm Röntgen" (Doc 1) and the full name "Wilhelm Conrad Röntgen" (Doc 3). Despite having access to the complete answer, the model outputs only "Wilhelm Röntgen", which fails under Exact Match evaluation.

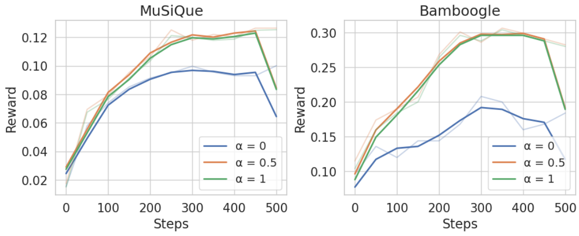

Figure 7: **Hyperparameter analysis of α in FOLR.** Results show that any $\alpha > 0$ effectively mitigates intra-group homogenization, with negligible difference between $\alpha = 0.5$ and $\alpha = 1$.

Critically, both trajectories successfully acquire the ground-truth information during retrieval, yet are penalized identically to trajectories that never retrieve relevant evidence. This exemplifies **process-level reward homogenization**: diverse intermediate successes are erased by a single binary outcome signal. TSPO addresses this by assigning partial reward to the turn where a_{gold} first appears, preserving signals of effective search behavior even when final synthesis is imperfect.

Table 7: Two failure trajectories where the ground-truth answer appears in retrieved documents (highlighted in red), but the final output is incorrect due to reasoning errors (top) or incomplete answer formatting (bottom). Both receive zero reward under outcome-level supervision despite successful evidence retrieval.

Question: name the tissue type of the inner most layer of cells?
Answer: Epithelium
Qwen2.5-7b-ins:
<think> he question is asking about the tissue type of the innermost layer of cells. This typically refers to the lining of an organ or a body cavity. However, without more context, it's difficult to determine which specific organ or body part is being referred to. I will conduct a search to find the most common innermost tissue type. </think>
<tool_call> {"name": "search", "arguments": {"query_list": ["innermost layer of cells tissue type"]}} </tool_call>
<tool_response> Doc 1 (Title: Epithelium)Epithelium Epithelium () is one of the four basic types of animal tissue, along with connective tissue, muscle tissue and nervous tissue. Epithelial tissues line the outer surfaces of organs and blood vessels throughout the body, as well as the inner surfaces of cavities in many internal organs... Doc 2: (Title: Epithelium)form is epithelia... There are three principal shapes of epithelial cell: squamous, columnar, and cuboidal. These can be arranged in a single layer of cells as simple epithelium, either squamous, columnar, or cuboidal, or in layers of two or more cells..
Doc 3: (Title: Periosteum)Periosteum The periosteum is a membrane that covers the outer surface of all bones, except at the joints of long bones... Doc 4: (Title: Anatomy)connective tissue in the extracellular matrix secreted by the epithelial cells. There are many different types of epithelium, modified to suit a particular function... Doc 5: (Title: Endoderm)Endoderm Endoderm is one of the three primary germ layers in the very early embryo. The other two layers are the ectoderm (outside layer) and mesoderm (middle layer), with the endoderm being the innermost layer... </tool_response>
<think> Based on the search results, the innermost layer of cells is typically referred to as the endoderm. The endoderm is one of the three primary germ layers in the very early embryo, along with the ectoderm and mesoderm. It forms the epithelial lining of multiple systems. Therefore, the tissue type of the innermost layer of cells is:<answer> Endoderm </answer> </think>
<answer> Endoderm </answer>

Question: who got the first nobel prize in physics?
Answer: Wilhelm Conrad Röntgen
Qwen2.5-7b-ins:
<think> Let me search for the information about the first Nobel Prize in Physics winner. </think>
<tool_call> {"name": "search", "arguments": {"query_list": ["first Nobel Prize in Physics winner"]}} </tool_call>
<tool_response> Doc 1 (Title: "Nobel Prize in Physics")The Nobel Prize in Physics () is a yearly award given by the Royal Swedish Academy of Sciences for those who have made the most outstanding contributions for mankind in the field of physics>... The first Nobel Prize in Physics was awarded to physicist Wilhelm Röntgen in recognition of the extraordinary services... Doc 2: (Title: "Nobel Prize in Physics")receive a diploma, a medal and a document confirming the prize amount... Doc 3: (Title: "Wilhelm Röntgen")...Wilhelm Conrad Röntgen (; ; 27 March 1845 – 10 February 1923) was a German mechanical engineer and physicist ... earned him the first Nobel Prize in Physics in 1901... </tool_response>
<think> Based on the search results, the first Nobel Prize in Physics was awarded to Wilhelm Röntgen in recognition of the extraordinary services he rendered by his discovery of X-rays. Therefore, the answer is <answer>Wilhelm Röntgen</answer>. </think>
<answer> Wilhelm Röntgen </answer>