Agentic Multi-Turn Reasoning: A Fairness Approach

Thanh-Dat Truong[1], **Sankalp Pandey**[1], **Hugh Churchill**[2], **Jackson Cothren**[3]
Marios Savvides[4], **Khoa Luu**[1]
[1]CVIU Lab, University of Arkansas, USA [2]Dep. of Physics, University of Arkansas, USA
[3]Dep. of Geosciences, University of Arkansas, USA [4]Carnegie Mellon University, USA
{tt032, sankalpp, hchurch, jcothre, khoaluu}@uark.edu, marioss@andrew.cmu.edu
https://uark-cviu.github.io/projects/Phi-MPO

Abstract

Recent advances in Large Language Models (LLMs) have enabled agentic systems to solve complex tasks through multi-turn planning, tool use, verification, and memory updates. However, learning agentic systems remains difficult due to two fundamental challenges, i.e., (1) long-horizon credit assignment, where supervision is available only at the final outcome, and (2) imbalanced data distributions, where dominant data patterns bias optimization and weaken adaptation to rare but informative reasoning behaviors. In this paper, we propose Fair Multi-Level Preference Optimization (Fair-MPO or Φ-MPO), a new preference optimization framework for agentic learning. We first show that Multi-Level Preference Optimization provides a principled and more computationally efficient framework for long-horizon reasoning. Then, we introduce a Fair Multi-Level Objective that addresses imbalance in agentic learning. We provide a comprehensive theoretical analysis demonstrating that our approach addresses both long-horizon reasoning and data imbalance. Our experiments on agentic reasoning benchmarks demonstrate that our approach achieves State-of-the-Art (SOTA) performance.

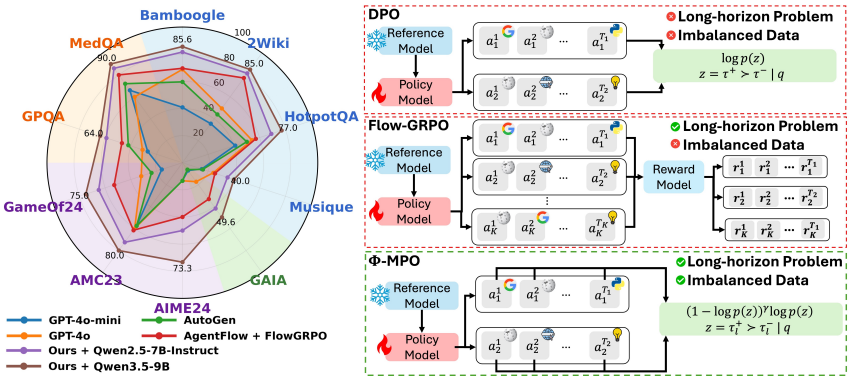

Figure 1: **Overview of the proposed Φ-MPO approach.** Left: Φ-MPO consistently outperforms GPT-4o, AgentFlow + FlowGRPO across four benchmarks. Right: Compared with DPO and Flow-GRPO, Φ-MPO addresses both long-horizon reasoning and imbalanced problems.

1 Introduction

The success of Large Language Models (LLMs) [1, 6, 52] has motivated a new class of *agentic* systems that reason over multiple steps, interact with external tools, maintain intermediate memory, and refine decisions based on feedback [20, 19, 22, 10]. Unlike single-turn generation, agentic reasoning is a multi-turn process in which the model must determine *when* to plan, *when* to call tools, *how* to update memory, and *how* to recover from mistakes, with each action shaping subsequent reasoning states and

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

final results [19, 22, 45, 18]. However, most existing agentic systems remain training-free, relying on prompting heuristics, handcrafted coordination rules, or static orchestration strategies [55, 13], which limits their adaptability to new environments, tools, and task distributions. Therefore, reliable and scalable multi-turn reasoning requires a **learning paradigm for agentic systems** that optimizes reasoning trajectories, tool usage, and module coordination based on interaction outcomes (Figure 1).

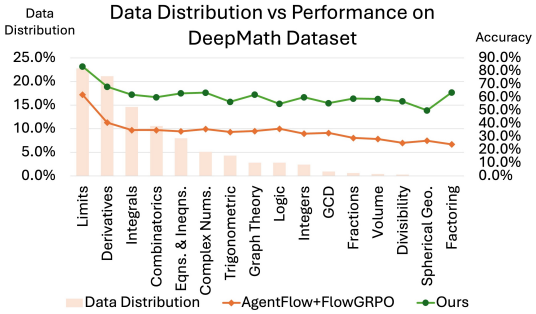

Figure 2: The Influence of Imbalance Data.

In learning agentic systems, two fundamental challenges arise: the **long-horizon problem** and the **imbalance problem**. The long-horizon problem arises because agentic reasoning unfolds over many interdependent steps, while supervision is typically available only at the final outcome [22]. Thus, early mistakes can propagate through planning, tool use, and memory updates, making it difficult to assign credit to intermediate decisions. Meanwhile, agentic learning often relies on imbalanced reasoning data, e.g., DeepMath [11], where topics, task types, difficulty levels, and reasoning patterns appear at highly uneven frequencies. As shown in Figure 2, this imbalance directly affects model performance. These challenges are amplified by the structured nature of agentic rollouts. Unlike single-output generation, an agentic rollout is a trajectory $\tau = (state_1, action_1, \ldots, state_T, action_T)$, where each state depends on prior reasoning, tool feedback, and memory updates. Thus, data-level imbalance propagates into trajectory-level imbalance: dominant topics and familiar tool-use patterns produce many redundant trajectories, while successful trajectories for rare or difficult cases remain scarce. This skew is further compounded by error propagation, e.g., an incorrect search query may retrieve irrelevant evidence, mislead later reasoning, and ultimately produce an incorrect answer. Therefore, a principled learning framework must address both long-horizon credit assignment and the propagation of imbalance from data to trajectories.

Recent studies have made important progress, but remain inadequate under long horizons and imbalanced data. Outcome-based reinforcement learning (RL) methods [44, 62, 45] and in-the-flow optimization frameworks [22] optimize trajectory generation from final rewards, yet often suffer from high variance, unstable optimization, and weak credit assignment when rewards are sparse. Meanwhile, Proximal Policy Optimization (PPO) [42], Group Relative Policy Optimization (GRPO) [44], or Flow-GRPO [22] improve policy learning from interaction trajectories, with GRPO using group-relative normalization and Flow-GRPO propagating final rewards across intermediate rollout steps. However, credit assignment remains coarse because the same outcome reward is shared across many decisions that differ in causal importance. Moreover, when successful rollouts are rare, optimization can be dominated by unsuccessful or low-quality trajectories, leading to biased updates and weak recovery from early mistakes. Direct Preference Optimization (DPO) provides a more stable alternative, learning from pairwise comparisons without explicit reward modeling or policy-gradient updates [40]. However, standard DPO is designed for single-output preference and does not explicitly model the sequential structure of agentic trajectories. Since DPO relies on informative and reasonably balanced preference pairs, it is also vulnerable to the highly skewed distributions produced by agentic rollouts. Thus, neither existing RL-based methods nor standard DPO fully addresses the dual challenge of long horizon and imbalance in agentic learning.

Contributions. In this work, we propose **Fair Multi-Level Preference Optimization** (**Fair-MPO** or **Φ-MPO**) for learning multi-turn agentic reasoning with long-horizon feedback and an imbalanced data distribution. Our contributions can be summarized as follows. First, we introduce a new **Multi-level Preference Learning** (MPO) framework for agentic systems that models each rollout as a structured reasoning trajectory and optimizes the policy through both trajectory-level and state-level supervision. Second, we identify the **imbalance problem in agentic learning** followed by developing a new **Φ-MPO learning objective** that improves robustness to skewed distributions. Third, we provide a theoretical analysis from two complementary perspectives: (1) we first show that DPO shares the same pairwise optimization direction as GRPO, yielding a more computationally efficient alternative. and (2) we then show that our fair multi-level learning objective mitigates optimization bias under imbalanced data. Finally, extensive experiments and ablation studies on standard benchmarks demonstrate that our approach improves final reasoning accuracy over prior

methods. Our results establish Φ-MPO as an effective learning framework for agentic systems in dynamic multi-turn environments.

2 Related Work

Agentic Learning. Recent studies formulates tool-integrated reasoning as a RL problem, where a LLM learns to interleave reasoning with external actions. Most methods adopt a monolithic policy that decides when to reason, when to invoke tools, and how to incorporate feedback. This paradigm has shown strong results in mathematical reasoning [31, 57, 9, 21, 18, 52, 49, 50, 26], web-based question answering [3, 19, 45, 20, 46], and multi-tool environments through data synthesis, unified training pipelines, and reward design [8, 17, 37, 65]. However, as task complexity and planning horizon increase, monolithic optimization faces severe long-horizon credit assignment, since final outcomes must be attributed to many intermediate reasoning steps, routing decisions, and tool calls [63, 54]. This can lead to unstable optimization, including inefficient tool use [53, 38], weak personalization [4], and poor alignment with user preferences [15]. Recent agentic systems offer an alternative by decomposing reasoning into specialized modules or agents for planning, execution, verification, and coordination. Many systems are training-free and orchestrate pretrained LLMs through prompting, role assignment, or handcrafted control logic [55, 13, 28]. While modularity improves flexibility, static orchestration limits adaptation because collaboration strategies are prescribed rather than learned from interaction. Recent methods train agentic systems for improved coordination and tool use [7, 23], but often rely on offline supervised fine-tuning or preference optimization over static datasets [33, 35]. Thus, training remains disconnected from live multi-turn agent-tool dynamics, limiting adaptation to evolving observations, recovery from early mistakes, and plan revision over long horizons.

Preference Learning and Policy Optimization. Preference learning and policy optimization are widely used to align LLMs with human or task-level feedback [66, 5]. Standard Reinforcement Learning from Human Feedback (RLHF) trains a reward model from ranked outputs and then optimizes the policy with PPO [42, 34]. DPO [40, 51, 27] instead directly optimizes preference pairs, avoiding explicit reward modeling. For reasoning, GRPO [44] uses group-relative comparisons to stabilize policy optimization, while Flow-GRPO [22] extends on-policy optimization to long-horizon agentic rollouts with sparse trajectory-level rewards. Despite this progress, RL methods still provide coarse supervision under delayed feedback, and standard preference optimization remains designed for single-output comparisons rather than structured, imbalanced, and multi-step agentic trajectories.

3 The Proposed Fair Multi-level Preference Optimization (Φ-MPO) Approach

In this section, we propose a new learning approach that enables the optimization of agentic systems within a multi-turn reasoning loop. Formally, given an input query q and a predefined toolset $\mathcal{K}$, our goal is to learn a policy that composes tool-usage actions to solve the query through iterative interaction. Following [22, 19], the agentic reasoning is decomposed into four modules: **Action Planner** $\mathcal{P}$, **Tool Executor** $\mathcal{E}$, **Execution Verifier** $\mathcal{V}$, and **Solution Generator** $\mathcal{G}$. These modules communicate through a shared evolving memory $\mathcal{M}$, which stores intermediate reasoning traces, tool outputs, and verification feedback. The overall process is formulated as a multi-turn Markov Decision Process (MDP). Let $\mathcal{M}^t$ denote the memory before turn t, with $\mathcal{M}^1$ initialized from q. At each turn, the planner $\mathcal{P}$ is modeled as a trainable policy π_θ that samples an action $a^t \sim \pi_\theta(a^t \mid q, \mathcal{K}, \mathcal{M}^t)$, where a^t specifies the current reasoning decision, including sub-goal generation, tool selection, and memory-conditioned retrieval. Conditioned on a^t, the executor invokes the selected tool and returns an observation $e^t \sim \mathcal{E}(e^t \mid a^t, \mathcal{K})$. The verifier then produces a binary signal $v^t \sim \mathcal{V}(v^t \mid q, e^t, \mathcal{M}^t)$ indicating whether the accumulated evidence is sufficient to generate the final answer. If $v^t = 0$, the memory is updated as $\mathcal{M}^{t+1} = f_{\text{mem}}(\mathcal{M}^t, a^t, e^t, v^t)$, providing an explicit and controllable representation of intermediate reasoning states. The process continues until $v^t = 1$ or the maximum number of turns is reached. At the final turn T, the solution genera tor produces the output $o \sim \mathcal{G}(o \mid q, \mathcal{M}^T)$, conditioned on both the query and the accumulated reasoning evidence. Then, the multi-turn reasoning process defines a structured trajectory $\tau = \{(a^t, e^t, v^t)\}_{t=1}^{T}$, which records the sequence of planning decisions, tool interactions, and verification signals as in Eqn. (1).

$$\pi_\theta(\tau, o \mid q) = \left[\prod_{t=1}^{T} \pi_\theta(a^t \mid q, K, \mathcal{M}^t)\, \mathcal{E}(e^t \mid a^t, K)\, \mathcal{V}(v^t \mid q, e^t, \mathcal{M}^t) \right] \mathcal{G}(o \mid q, \mathcal{M}^t). \tag{1}$$

Similar to [22], Eqn. (1) decomposes complex reasoning into a sequence of observable transitions, allowing the reasoning trajectory to be explicitly represented through structured intermediate states. The planner π_θ is optimized within the reasoning loop, enabling direct improvement of intermediate decisions and more fine-grained credit assignment across multi-turn, tool-integrated trajectories.

Agentic Learning. The Action Planner policy π_θ can be optimized via Reinforcement Learning from Human Feedback (RLHF). In particular, let $R(\tau)$ be the reward model to evaluate the correctness and efficiency of a complete trajectory τ. The planner can be optimized as follows:

$$\pi_\theta^\star \;=\; \underset{\pi_\theta}{\operatorname{argmax}}\; \mathbb{E}_{q\sim\mathcal{Q}}\mathbb{E}_{\tau\sim\pi_\theta(\tau|q)|}\big[R(\tau)\big] \quad \text{s.t.} \quad D_{\mathrm{KL}}\big(\pi_t(\cdot \mid q)\,\big\|\,\pi_{ref}(\cdot \mid q)\big) \le \delta, \tag{2}$$

where π_{ref} is the reference policy, $D_{\mathrm{KL}}(\pi_\theta\|\pi_{ref})$ is the KL divergence to measure the difference between two policies, and τ is a trajectory rollout. In multi-turn reasoning, the contribution of each intermediate action a^t is difficult to observe directly, since its effect may only appear after several subsequent steps. Thus, defining step-wise rewards can introduce noisy or inconsistent supervision. Following [22], we instead adopt a trajectory-level reward, where all actions in a rollout share a common signal determined by the correctness of the final solution, i.e., $r = R(a^t) = \bar{R}(o, q, y^*)$ ($\forall t = 1, \ldots, T$), where y^* is the ground-truth answer, and $\bar{R}(o, q, y^*) \in \{0, 1\}$ evaluates whether the final prediction o is correct. Similar to [22], we adopt the LLM-as-judge protocol for the reward function $\bar{R}(\cdot)$. This approach ensures a global correctness signal throughout the reasoning trajectory, encouraging each intermediate decision a^t to contribute to producing a correct final solution.

Challenges and Limitations of Prior Agentic Learning. Technically, multi-turn agentic learning must handle both long-horizon credit assignment and imbalanced data distributions. Given a trajectory $\tau = \{a^t\}_{t=1}^T$, the final output depends on the accumulated memory $\mathcal{M}^T$, so the effect of an early action may only appear after several planning, execution, and memory-update steps. Meanwhile, an imbalance in topics, tasks, and reasoning patterns propagates into rollouts, causing dominant trajectories or tool-use behaviors to receive disproportionate gradient updates. While Eqn. (2) can be optimized with PPO [42], PPO requires explicit reward modeling and value estimation over long trajectories, which can be biased and unstable under final-outcome supervision. GRPO [22] mitigates value estimation via group-relative comparison, but still requires expensive on-policy sampling over multiple long rollouts involving tool calls, verification, and memory updates. These limitations motivate a Direct Preference Optimization approach that avoids explicit reward and value modeling while providing more stable learning under long-horizon dependency and imbalanced data.

3.1 Agentic Learning via Preference Optimization

The constrained optimization problem in Eqn. (2) can be rewritten using the Lagrangian [40] form as:

$$\pi_\theta^\star = \underset{\pi_\theta}{\operatorname{argmax}}\; \mathbb{E}_{q\sim\mathcal{Q}}\mathbb{E}_{\tau\sim\pi_\theta(\tau|q)}\big[R(\tau)\big] - \beta D_{\mathrm{KL}}\Big(\pi_\theta(\cdot \mid q)\,\Big\|\,\pi_{\mathrm{ref}}(\cdot \mid q)\Big), \tag{3}$$

where β is the Lagrangian multiplier that controls the deviation of the current policy from the reference model. Then, the optimal policy of Eqn. (3) [40] admits the following closed form:

$$\pi_\theta^\star(\tau \mid q) = \frac{\pi_{\mathrm{ref}}(\tau \mid q)\exp\Big(\frac{1}{\beta}R(\tau)\Big)}{Z(q)}, \qquad R(\tau) = \beta\log\frac{\pi_\theta^\star(\tau \mid q)}{\pi_{\mathrm{ref}}(\tau \mid q)} + \beta\log Z(q), \tag{4}$$

where $Z(q) = \sum_\tau \pi_{\mathrm{ref}}(\tau \mid q)\exp\Big(\frac{1}{\beta}R(\tau)\Big)$ is the partition function.

Pairwise Trajectory Preferences. For each query q, we denote by τ^+ a preferred trajectory that yields a more effective reasoning process or a correct final solution, and by τ^- a dispreferred trajectory that produces inferior reasoning or an incorrect solution. Then, the preference of τ^+ over τ^- conditioned on query q can be modeled using the Bradley-Terry model, i.e., $p(\tau^+ \succ \tau^- \mid q) = \sigma\Big(\beta\big[R(\tau^+) - R(\tau^-)\big]\Big)$, where $\sigma(u)$ is the sigmoid function. Accordingly, maximizing the conditional likelihood over pairwise trajectory preferences can be written as the following objective:

$$\mathcal{L}_{\mathrm{pref}} = \mathbb{E}_{(q,\tau^+,\tau^-)}\Big[-\log\sigma\Big(\beta\big[R(\tau^+) - R(\tau^-)\big]\Big)\Big]. \tag{5}$$

Agentic Learning Paradigm with Trajectory-level DPO. Under the optimality condition in Eqn. (4), the reward difference in Eqn. (5) can be rewritten in terms of trajectory-level policy log-ratios. Therefore, the preference objective can be reformulated as the following **Trajectory-level DPO** loss:

$$\mathcal{L}_{\mathrm{DPO}}(\pi_\theta, \pi_{\mathrm{ref}}) = -\mathbb{E}_{q,\tau^+,\tau^-}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(\tau^+ \mid q)}{\pi_{\mathrm{ref}}(\tau^+ \mid q)} - \beta\log\frac{\pi_\theta(\tau^- \mid q)}{\pi_{\mathrm{ref}}(\tau^- \mid q)}\right)\right]. \tag{6}$$

Since each trajectory τ^+ or τ^- is generated through a multi-turn reasoning process, its likelihood can be factorized over the sequence of intermediate actions as Eqn. (1).

Interpretation. The Trajectory-level DPO optimizes the planner to favor trajectories with better tool use and more accurate final solutions, while staying close to a reference policy for stability. Specifically, Eqn. (6) increases the relative log-likelihood of preferred trajectories τ^+ over dispreferred ones τ^- compared to the reference policy. Unlike RLHF methods that require explicit reward modeling or value estimation, this formulation learns directly from pairwise trajectory preferences, making it suitable for multi-turn reasoning where step-wise rewards are difficult to define.

Limitations of Trajectory-level DPO. Although Trajectory-level DPO enables stable optimization without explicit reward modeling, it still provides only coarse supervision for multi-turn reasoning. Since the preference signal is defined by the final outcome, all intermediate actions $\{a^t\}_{t=1}^T$ receive the same supervision regardless of their individual contributions. This makes it difficult to identify which reasoning steps cause success or failure, leading to inefficient policy improvement and high-variance optimization when trajectories share similar final outcomes but differ in intermediate quality.

3.1.1 Multi-level Preference Learning Approach to Agentic Learning

To address the long-horizon problem aforementioned, we propose a new Multi-level Preference Learning (MPO) that decomposes trajectory-level preferences into multiple aligned state-level comparisons. Let a reasoning trajectory be denoted as $\tau = \{(s^t, a^t)\}_{t=1}^T$, where each state $s^t = (q, K, \mathcal{M}^t)$ consists of the query q, toolset K, and evolving memory $\mathcal{M}^t$ that summarizes the accumulated reasoning context up to step t. Since the preferred τ^+ and dispreferred trajectories and τ^- may have different lengths and reasoning structures, we define an alignment set $A \subseteq \{1, .., T^+\} \times \{1, .., T^-\}$ (T^+ and T^- are the lenghts of τ^+ and τ^-) that pairs intermediate reasoning states according to their semantic similarity, enabling comparison between corresponding reasoning steps across trajectories. For each aligned pair $(i, j) \in A$, we define the preference probability as follows:

$$p_{ij} = \sigma\Big(\beta\big[\log \pi_\theta(a_i^+|s_i^+) - \log \pi_\theta(a_j^-|s_j^-)\big] - \beta\big[\log \pi_{\text{ref}}(a_i^+|s_i^+) - \log \pi_{\text{ref}}(a_j^-|s_j^-)\big]\Big). \tag{7}$$

Then, our proposed **Multi-level Preference Optimization** objective can be defined as follows:

$$\mathcal{L}_{\text{MPO}}(\pi_\theta, \pi_{\text{ref}}) = -\mathbb{E}_{(\tau^+,\tau^-)} \frac{1}{|A|} \sum_{(i,j)\in A} \log p_{ij}. \tag{8}$$

Unlike Trajectory-level DPO, our MPO objective introduces multiple intermediate preference constraints across aligned reasoning states. This formulation enables more effective credit assignment across multi-turn reasoning steps, allowing the policy to identify which intermediate decisions contribute positively or negatively to the final outcome. Consequently, MPO mitigates the long-horizon dependency in agentic learning and provides a more stable optimization signal across trajectories with varying reasoning structures and lengths.

Rollout Scoring for On-the-Fly Preference Construction. For each query q, we generate a small set of K rollouts $\{\tau_k\}_{k=1}^K$ from the current policy π_θ and score each trajectory by both final correctness and reasoning efficiency. Let $R(\tau_k)$ denote the final reward, $U(\tau_k)$ the number of tool calls, and T_k the trajectory length. We define $S(\tau_k) = R(\tau_k) - 0.05\frac{U(\tau_k)}{U_{\max}} - 0.05\frac{T_k}{T_{\max}}$, where $U_{\max}$ and $T_{\max}$ are computed within the current rollout group. The first term with a higher coefficient prioritizes trajectories that produce correct final solutions, while the second and third terms favor trajectories that solve the query using fewer tool invocations and fewer reasoning steps. We then select the highest-scored rollout as τ^+ and the lowest-scored rollout as τ^-, enabling on-the-fly preference construction that encourages correct and efficient agentic reasoning.

Alignment Set A Construction. Preferred and dispreferred trajectories may have different lengths and reasoning structures, so step-wise correspondence is not directly available. We therefore construct an alignment set $A \subseteq \{1, \ldots, T^+\} \times \{1, \ldots, T^-\}$ that pairs semantically related states. Each state $s^t = (q, a^t, k^t, \mathcal{M}^t)$ includes the planner action, selected tool, and evolving memory. For efficiency, we use monotonic dynamic programming alignment based on structured state similarity. For each preferred state s_i^+, we select:

$$j^*(i) = \arg \max_{j\in\{1,\ldots,T^-\}} \text{sim}(s_i^+, s_j^-), \qquad \text{s.t. } j^*(i+1) \geq j^*(i). \tag{9}$$

The similarity is computed as $\text{sim}(s_i^+, s_j^-) = \mathbf{1}(a_i^+ = a_j^-) + \mathbf{1}(k_i^+ = k_j^-) + \text{sim}_m(\mathcal{M}_i^+, \mathcal{M}_j^-)$ where sim_m is the Jaccard overlap between tokenized memory states. The final alignment set

is $A = \{(i, j^*(i))\}_{i=1}^{T^+}$, with monotonic filtering to retain at most $\min(T^+, T^-)$ aligned pairs. This construction preserves temporal consistency with low overhead, allowing MPO to propagate preference signals to intermediate reasoning states.

3.1.2 Theoretical Analysis of Preference Learning in Agentic Learning

Recent work shows that GRPO provides stable learning signals for multi-turn agentic reasoning through intra-group relative advantages [22]. Following prior interpretations of DPO and GRPO as contrastive objectives [56], we show that K-GRPO can be decomposed into a positively weighted combination of pairwise preference gradients. The, we prove that our preference learning induces an equivalent optimization direction as GRPO while avoiding explicit value estimation. For simplicity, we present the equivalence at the trajectory level since our MPO preserves the same pairwise structure while extending supervision to intermediate reasoning states. Let $\mathcal{G}(q) = \{\tau_1, \ldots, \tau_K\}$ where $\tau_i \sim \pi_\theta(\cdot|q)$ be a group of $K \geq 2$ sampled trajectories. Then, we define preferred $\mathcal{P}(q)$ and dispreferred $\mathcal{N}(q)$ trajectory sets, $m = |\mathcal{P}(q)|$, $n = |\mathcal{N}(q)|$, $m + n = K$, and assume non-degenerate groups ($1 \leq m \leq K - 1$). The trajectory-level score function can be defined as $s_\theta(q, \tau) = \nabla_\theta \log \pi_\theta(\tau|q)$.

Theorem 1 (Pairwise Equivalence of DPO and K-GRPO up to Positive Weighting) *Under unclipped trajectory-level updates with binary rewards, the K-GRPO gradient can be written as*

$$g_\theta^{\text{K-GRPO}}(q) = \alpha(q)\, \mathbb{E}_{\tau^+ \sim \text{Unif}(\mathcal{P}(q)),\, \tau^- \sim \text{Unif}(\mathcal{N}(q))} \Big[s_\theta(q, \tau^+) - s_\theta(q, \tau^-) \Big]. \tag{10}$$

where the group-dependent positive weight is $\alpha(q) = \sqrt{\hat{p}_q(1 - \hat{p}_q)} = \frac{\sqrt{mn}}{K}$, $\hat{p}_q = \frac{m}{K}$. Moreover, the Trajectory-level DPO gradient can be written as $\nabla_\theta L_{\text{DPO}} = -\mathbb{E}_{q,\tau^+,\tau^-} \Big[w_\theta(q, \tau^+, \tau^-) \big(s_\theta(q, \tau^+) - s_\theta(q, \tau^-) \big) \Big]$, s where $w_\theta(q, \tau^+, \tau^-) = \beta\sigma(-\Delta_\theta(q, \tau^+, \tau^-)) > 0$ with $\Delta_\theta(q, \tau^+, \tau^-)$ is the DPO logit margin. Therefore, K-GRPO and DPO share the same pairwise gradient basis $s_\theta(q, \tau^+) - s_\theta(q, \tau^-)$ and differ only by positive weighting (the proof is provided in the appendix).

Interpretation. Theorem 1 shows that GRPO implicitly performs pairwise preference learning over preferred-dispreferred trajectory pairs, while DPO makes this structure explicit through adaptive logistic weighting and reference-policy regularization. Thus, DPO preserves the same optimization direction as GRPO but avoids explicit value estimation and repeated on-policy sampling, yielding a more efficient framework for long-horizon agentic reasoning.

3.2 Imbalanced Learning in Agentic Reasoning

In multi-turn agentic reasoning, data imbalance propagates through rollout generation and further induces imbalance in the aligned state pairs A. Since π_θ frequently generates common reasoning patterns, e.g., repeated tool selections or standard intermediate steps, sampled rollouts contain many similar trajectory segments. As a result, alignment construction produces many *easy* pairs where the preferred action already has high probability, while rare but critical transitions, e.g., correcting an incorrect tool call or revising evidence, yield only a few *hard* pairs requiring meaningful policy updates. Thus, A becomes dominated by easy-majority pairs, biasing gradients toward frequent reasoning structures and under-optimizing rare but important decisions. Formally, under Eqn. (8), the gradient magnitude of each aligned pair is proportional to $\psi_0(p) = \left| \frac{\partial(-\log p_{i,j})}{\partial z} \right| = 1 - p_{i,j}$, where z is the logit margin. Although easy pairs with $p_{i,j} \approx 1$ have small individual gradients, their large quantity can still dominate the aggregate update. This leads to biased optimization and slow improvement on underrepresented hard decisions.

Fair Multi-level Preference Optimization. To mitigate the bias caused imbalance data, inspired by [25, 51], we introduce a focal weighting mechanism that down-weights easy preference pairs and emphasizes hard pairs. Formally, we formulate our **Φ-MPO** objective as follows:

$$\mathcal{L}_{\text{FairMPO}} = -\mathbb{E}_{(\tau^+,\tau^-)} \frac{1}{|A|} \sum_{(i,j)\in A} (1 - p_{i,j})^\gamma \log p_{i,j}, \tag{11}$$

where $\gamma \geq 0$ is the focal parameter. Compared to Eqn. (8), the factor $(1 - p_{i,j})^\gamma$ suppresses the contribution of easy pairs ($p_{i,j} \approx 1$) and increases the importance of hard pairs with smaller $p_{i,j}$.

Theorem 2 (Suppression of Easy-Majority Dominance in Φ-MPO) *Let us define* $\mathcal{L}_\gamma(p) = -(1-p)^\gamma \log p$, $p = \sigma(z)$, $\gamma \geq 0$. *Then, the margin-gradient magnitude can be formed as* $\psi_\gamma(p) := \left|\frac{\partial \mathcal{L}_\gamma(p)}{\partial z}\right|$, *Suppose that, at level* ℓ, *the aligned pairs are partitioned into an easy-majority set* $E^{(\ell)}$ *and a hard-minority set* $H^{(\ell)}$ *such that* $p_{ij}^{(\ell)} \geq 1-\varepsilon \quad \forall (i,j) \in E^{(\ell)}$ *and* $p_{ij}^{(\ell)} \leq 1-\delta \quad \forall (i,j) \in H^{(\ell)}$ *for some* $0 < \varepsilon < \delta < 1$. *Then*

$$\sum_{(i,j)\in E^{(\ell)}} \psi_\gamma\left(p_{ij}^{(\ell)}\right) \leq |E^{(\ell)}|(\gamma+1)\varepsilon^{\gamma+1} \quad \text{and} \quad \sum_{(i,j)\in H^{(\ell)}} \psi_\gamma\left(p_{ij}^{(\ell)}\right) \geq |H^{(\ell)}|\delta^{\gamma+1}$$
$$\Rightarrow \frac{\sum_{(i,j)\in E^{(\ell)}} \psi_\gamma\left(p_{ij}^{(\ell)}\right)}{\sum_{(i,j)\in H^{(\ell)}} \psi_\gamma\left(p_{ij}^{(\ell)}\right)} \leq \frac{|E^{(\ell)}|}{|H^{(\ell)}|}(\gamma+1)\left(\frac{\varepsilon}{\delta}\right)^{\gamma+1}. \tag{12}$$

Therefore, Φ-MPO suppresses the optimization influence of overrepresented easy pairs and mitigates imbalance caused by easy-majority dominance (the proof is provided in the appendix).

Interpretation. When $\gamma = 0$, Φ-MPO recovers standard MPO, whose easy-majority versus hard-minority gradient contribution scales with the frequency ratio $|E^{(\ell)}|/|H^{(\ell)}|$. As γ increases, easy-pair contributions decay at the higher-order rate $\varepsilon^{\gamma+1}$, while hard pairs retain meaningful gradients. Thus, Φ-MPO prevents frequent easy decisions from dominating optimization, yielding fairer updates and better adaptation to rare but informative reasoning transitions.

Final Agentic Learning Objective. In our framework, we combine trajectory-level and MPO to improve both global task success and intermediate reasoning quality. Trajectory-level DPO encourages trajectories with correct final outcomes, while MPO provides denser supervision over aligned reasoning states, improving tool use and evidence accumulation:

$$\mathcal{L}_{\text{Final}} = \lambda_{\text{DPO}}\mathcal{L}_{\text{DPO}} + \lambda_{\Phi-\text{MPO}}\mathcal{L}_{\Phi-\text{MPO}}, \tag{13}$$

where λ_{DPO} and $\lambda_{\Phi-\text{MPO}}$ balance the two losses. We also apply the focal weighting in Eqn. (11) to $\mathcal{L}_{\text{DPO}}$ to reduce easy-majority dominance. Figure 3 illustrates the overall framework.

4 Experiments

4.1 Implementation and Benchmarks

Implementation. We adopt the implementation of [22] and instantiate the *Action Planner*, *Tool Executor*, *Execution Verifier*, and *Solution Generator* with Qwen2.5-7B-Instruct [47]. The agent uses five tools: *Base Generator*, *Python Coder*, *Google Search*, *Wikipedia Search*, and *Web Search*. Following standard protocols [22, 19], we train with learning rate 10^{-6}, penalty $\beta = 0.001$, planner temperature 0.5, maximum planner length 2048, batch size 12, and $K = 4$ rollouts per input. We set the maximum number of turns to 3 during training and 10 during evaluation, use GPT-4o as the LLM judge for the reward function, and run all tool LLMs with temperature 0.0 for deterministic execution. Training is conducted on 12 NVIDIA L40S GPUs. We set $\gamma = 2.0$ and $\lambda_{\text{DPO}} = \lambda_{\Phi-\text{MPO}} = 1.0$. Detailed tool definitions and memory updates are provided in the Appendix.

Datasets and Benchmarks. Following prior protocols [19, 22], our framework is trained on a mixture of Search-R1 [19] and DeepMath [11]. For fair comparisons, we conduct our evaluation on

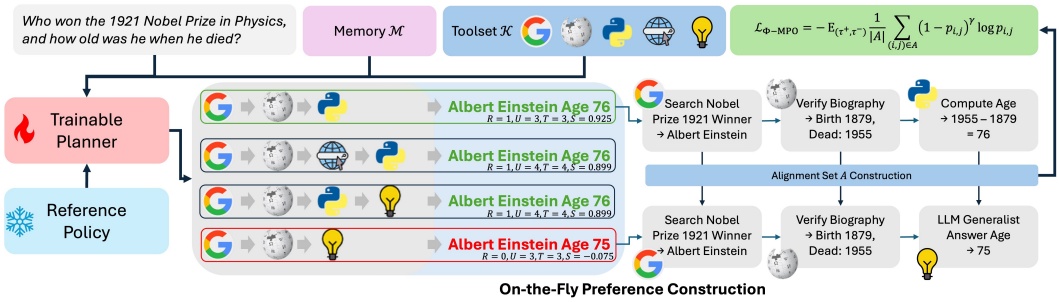

Figure 3: **Our Proposed Φ-MPO Framework**.

Table 1: **Accuracy Results on Search & Agentic Reasoning Benchmarks.**

Method	Model Size	Knowledge-Intensive Search				Agentic
		Bamboogle	2Wiki	HotpotQA	Musique	GAIA
Qwen2.5-7B-Inst	7B	12.0	23.0	21.0	6.0	3.2
Qwen2.5-14B-Inst	14B	21.6	26.7	20.0	8.0	5.5
Qwen2.5-32B-Inst	32B	24.0	26.7	27.0	6.0	9.5
Llama-3.3-70B-Inst	70B	18.4	22.7	52.0	16.0	3.2
GPT-4o-mini [16]	8B	40.8	35.6	41.0	15.0	7.1
GPT-4o [16]	200B	68.8	49.5	54.0	24.0	17.3
SFT	7B-Inst	12.0	25.9	22.0	6.6	3.2
Iter-RetGen [43]	7B-Inst	36.8	33.6	37.4	17.8	3.9
Search-R1 [19]	7B-Inst	43.2	38.2	37.0	14.6	19.1
ZeroSearch [46]	7B-Base	27.8	35.2	34.6	18.0	16.5
ReSearch [3]	7B-Base	42.4	47.6	43.5	22.3	17.3
StepSearch [54]	7B-Base	40.0	36.6	38.6	22.6	-
VerlTool [17]	7B-Base	46.4	45.3	44.8	19.3	11.2
AutoGen [55]	7B-Inst	59.6	44.0	50.0	15.9	6.3
AgentFlow [22]	7B-Inst	58.4	60.0	51.3	19.2	17.2
FlowGRPO [22]	7B-Inst	69.6	77.2	57.0	25.3	33.1
Φ-MPO	7B-Inst	**81.6**	**81.5**	**69.0**	**35.0**	**41.7**
Φ-MPO	Qwen3.5-9B	**85.6**	**85.0**	**77.0**	**40.0**	**49.6**

Table 2: **Accuracy Results on Math & Scientific Reasoning Benchmarks.** (7B-Inst: Qwen2.5-7B-Instruct, 7B-Base: Qwen2.5-7B-Base)

Method	Model Size	Mathematical Reasoning			Scientific Reasoning	
		AIME24	AMC23	GameOf24	GPQA	MedQA
Qwen2.5-7B-Inst	7B	6.7	47.5	33.0	34.0	66.0
Qwen2.5-14B-Inst	14B	6.7	60.0	25.0	31.0	75.0
Llama-3.3-70B-Inst	70B	6.7	47.5	31.0	35.0	67.0
Llama-3.1-405B-Inst	405B	26.7	47.5	23.0	30.0	62.0
GPT-4o-mini [16]	8B	13.3	57.5	16.0	27.0	66.0
GPT-4o [16]	200B	13.3	60.0	32.0	31.0	60.0
SFT	7B-Inst	6.7	47.5	33.0	34.0	66.0
SimpleRL [64]	7B-Base	16.7	60.0	33.0	45.0	65.0
Open-Reasoner [14]	7B-Base	16.7	54.9	32.0	34.0	54.0
General-Reasoner [29]	7B-Base	13.3	55.0	33.0	35.5	61.0
Luffy [58]	7B-Inst	30.7	44.8	33.0	34.0	77.0
TIR [59]	7B-Inst	10.0	50.0	33.0	42.0	76.8
ToRL [21]	7B-Inst	20.0	60.0	31.0	35.0	76.5
AutoGen [55]	7B-Inst	13.3	57.5	24.0	42.0	72.0
AgentFlow [22]	7B-Inst	16.7	47.4	31.0	37.0	76.0
Flow-GRPO [22]	7B-Inst	40.0	61.5	53.0	47.0	80.0
Φ-MPO	7B-Inst	**50.0**	**72.5**	**65.0**	**59.0**	**86.0**
Φ-MPO	Qwen3.5-9B	**73.3**	**80.0**	**75.0**	**64.0**	**90.0**

four benchmarks: (1) ***Knowledge-Intensive Search*** includes Bamboogle [36] , 2Wiki [12], HotpotQA [61], and Musique [48]; (2) ***Agentic Reasoning*** consists of GAIA [32]; (3) ***Mathematical Reasoning*** includes AIME2024 [2], AMC23 [30], and GameOf24 [24]. and (4) ***Scientific Reasoning*** includes GPQA [41] and MedQA [60].

4.2 Main Results

Search-Intensive and Agentic Benchmarks. Table 1 shows that our method consistently achieves the best performance across all knowledge-intensive search benchmarks and the agentic reasoning benchmark GAIA. With the Qwen2.5-7B-Instruct backbone, our method attains 81.6% on Bamboogle, 81.5% on 2Wiki, 69.0% on HotpotQA, 35.0% on Musique, and 41.7% on GAIA, consistently outperforming prior search-based, agentic, and reinforcement learning baselines, including Flow-GRPO [22]. Notably, our 7B model even surpasses GPT-4o across all five benchmarks, highlighting the effectiveness of our proposed framework for both knowledge-intensive retrieval and multi-turn agentic reasoning. When using the stronger Qwen3.5-9B [39] backbone, the performance is further improved to 85.6%, 85.0%, 77.0%, 40.0%, and 49.6%, respectively, which further confirms the robustness and scalability of our approach. Figure 4 shows a case study of our approach.

Mathematical and Scientific Reasoning Benchmarks. The results in Table 2 further demonstrate the strong effectiveness of our approach on both mathematical and scientific reasoning tasks. Using the Qwen2.5-7B-Instruct backbone, Φ-MPO achieves 50.0% on AIME24, 72.5% on AMC23, 65.0% on GameOf24, 59.0% on GPQA, and 86.0% on MedQA, establishing the best performance across all benchmarks. Compared prior Flow-GRPO [22], our approach obtains consistent gains on every task, with especially notable improvements on AIME24, GameOf24, and GPQA. These results suggest that our approach is particularly effective for challenging multi-step reasoning problems that require both precise planning and reliable tool use. When scaling to Qwen3.5-9B [39], the performance is further improved, which again confirms our robustness and scalability.

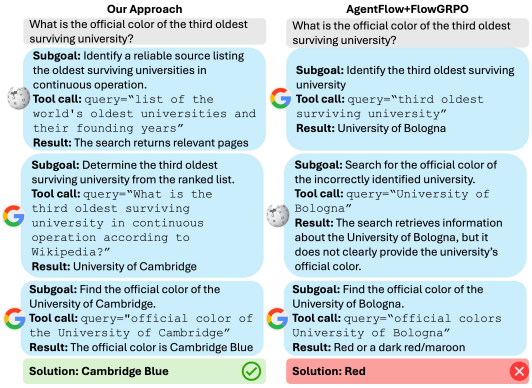

Figure 4: Our Case Study Example.

4.3 Ablation Studies

Effectiveness of Different Training Approaches. Table 3 compares Frozen, (Supervised Finetuning) SFT, Flow-GRPO, and our method. The Frozen baseline performs competitively on Bamboogle (58.4%) and 2Wiki (60.0%), but is limited on GAIA (17.2%) and AIME24 (16.7%). SFT performs worse across all benchmarks, suggesting that standard supervised tuning is insufficient for multi-turn agentic reasoning. Flow-GRPO improves performance to 69.6%, 77.2%, 33.1%, and 40.0%, respectively, confirming the benefit of reinforcement learning for long-horizon optimization. Our

method further outperforms Flow-GRPO on all benchmarks, demonstrating the effectiveness of preference optimization for agentic learning.

Effectiveness of Fair MPO. As shown in Table 3, Trajectory-level DPO achieves 61.6% on Bamboogle, 61.5% on 2Wiki, 21.3% on GAIA, and 30.0% on AIME24, outperforming Frozen and SFT baselines on most tasks. Adding MPO improves performance to 76.0%, 76.0%, 36.2%, and 43.3%, showing the benefit of intermediate flow supervision. Our final Φ-MPO achieves the best results, reaching 81.6%, 81.5%, 41.7%, and 50.0%, respectively. These results confirm that multi-level supervision and focal balancing both improve agentic preference learning under imbalanced data.

Table 3: Effectiveness of Our Proposed Objective.

DPO	Multi-Level	Fair	Bamboogle	2Wiki	GAIA	AIME24
✓			61.6%	61.5%	21.3%	30.0%
✓	✓		76.0%	76.0%	36.2%	43.3%
✓	✓	✓	**81.6%**	**81.5%**	**41.7%**	**50.0%**
Frozen			58.4%	60.0%	17.2%	16.7%
Supervised Fine-Tuning			30.4%	32.7%	6.3%	3.3%
Flow-GRPO			69.6%	77.2%	33.1%	40.0%

Effectiveness of Maximum Number of Turns. As shown in Table 4, increasing the turn budget consistently improves performance across all benchmarks. With $T = 3$, the model achieves only 60.8% on Bamboogle, 61.0% on 2Wiki, 21.3% on GAIA, and 26.7% on AIME24. Performance steadily improves as T increases to 5 and 7, with the best results at $T = 10$: 81.6%, 81.5%, 41.7%, and 50.0%, respectively. These results show that more interaction steps enable deeper planning, tool use, and recovery from early mistakes in long-horizon reasoning.

Table 4: Effectiveness of Number of Turns.

T	Bamboogle	2Wiki	GAIA	AIME24
3	60.8%	61.0%	21.3%	26.7%
5	69.6%	70.0%	29.9%	36.7%
7	76.8%	77.0%	37.0%	43.3%
10	**81.6%**	**81.5%**	**41.7%**	**50.0%**

Effectiveness of Focal Parameter γ. Table 5 studies the focal hyper-parameter γ, which controls the emphasis on hard reasoning decisions. When $\gamma = 0.0$, the objective reduces to standard MPO, achieving 76.0% on Bamboogle, 76.0% on 2Wiki, 36.2% on GAIA, and 43.3% on AIME24. Increasing γ from 0.5 to 2.0 consistently improves performance, with $\gamma = 2.0$ achieving the best results: 81.6%, 81.5%, 41.7%, and 50.0%, respectively. However, $\gamma = 5.0$ degrades performance, suggesting that overly strong focal weighting suppresses too many gradients and weakens learning. These results show that a moderate focal factor best balances easy-majority patterns and informative hard decisions.

Table 5: Effectiveness of Hyper-Parameter γ.

γ	Bamboogle	2Wiki	GAIA	AIME24
0.0	76.0%	76.0%	36.2%	43.3%
0.5	76.0%	76.5%	37.0%	46.7%
1.0	78.4%	78.0%	37.8%	46.7%
2.0	**81.6%**	**81.5%**	**41.7%**	**50.0%**
5.0	75.2%	75.5%	35.4%	43.3%

Generalization Across Model Scale. As in Table 6 , across Qwen2.5-3B-Instruct and Qwen2.5-7B-Instruct, our method consistently outperforms the Frozen and AgentFlow with FlowGRPO baselines on all benchmarks, showing robustness across model capacities. Notably, our Qwen2.5-3B variant already surpasses AgentFlow with Qwen2.5-7B on several tasks, achieving 72.0% on Bamboogle, 72.5% on 2Wiki, 32.3% on GAIA, and 40.0% on AIME24. Scaling to Qwen2.5-7B further improves performance to 81.6%, 81.5%, 41.7%, and 50.0%, respectively. The same trend holds for Qwen3.5, where Qwen3.5-9B outperforms Qwen3.5-4B and reaches 85.6% on Bamboogle, 85.0% on 2Wiki, 49.6% on GAIA, and 73.3% on AIME24. These results show that our method generalizes across backbone families and scales effectively with stronger models.

Table 6: Effectiveness of Model Scale.

Method	Size	Bamboogle	2Wiki	GAIA	AIME24
Frozen	Qwen2.5-3B-Instruct	53.6%	63.0%	14.3%	13.3%
	Qwen2.5-7B-Instruct	58.4%	60.0%	17.2%	16.7%
AgentFlow	Qwen2.5-3B-Instruct	68.8%	72.3%	29.1%	20.0%
	Qwen2.5-7B-Instruct	69.6%	77.2%	33.1%	40.0%
Ours	Qwen2.5-3B-Instruct	72.0%	72.5%	32.3%	40.0%
	Qwen2.5-7B-Instruct	**81.6%**	**81.5%**	**41.7%**	**50.0%**
Ours	Qwen3.5-4B	74.4%	74.0%	34.6%	33.3%
	Qwen3.5-9B	**85.6%**	**85.0%**	**49.6%**	**73.3%**

5 Conclusions and Limitations

Conclusions. This paper has presented a novel Φ-MPO approach to agentic learning. In particular, our proposed learning objective has been introduced to address both long-horizon and data imbalance problems in multi-turn agentic reasoning. Our theoretical analysis has also demonstrated the effectiveness and robustness of the proposed framework. Extensive experiments on multiple benchmarks have further confirmed the effectiveness of our method compared to other methods.

Limitations. Our paper adopts a set of design choices and hyper-parameters aligned with our theoretical analysis, but this also introduces several limitations, particularly in tuning the focal parameter γ, the weighting coefficients between trajectory-level and multi-level objectives, and the maximum rollout budget. In addition, the effectiveness of our approach relies on the quality of the

on-the-fly rollouts used to construct preference pairs, which may be sensitive to rollout diversity, reward reliability, and the stability of trajectory-level and step-level supervision. In long-horizon agentic settings, noisy tool feedback, imperfect final-outcome judgments, or insufficient exploration may lead to suboptimal preference signals and biased optimization. These limitations highlight the need for future studies on more robust preference optimization strategies for agentic learning.

Acknowledgment. This work is partly supported by DOW/ARO (No. W911NF261A114), NSF/IIS CAREER (No. 2442295), NSF/IIS (No. 2501021), NSF/BIO (No. 2524623), and NSF/OIA (No. 2445877). This research is also supported by the Arkansas High Performance Computing Center which is funded through multiple National Science Foundation grants and the Arkansas Economic Development Commission.

References

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] Art of Problem Solving. Aime problems and solutions, 2025.

[3] M. Chen, L. Sun, T. Li, H. Sun, Y. Zhou, C. Zhu, H. Wang, J. Z. Pan, W. Zhang, H. Chen, et al. Learning to reason with search for llms via reinforcement learning. *arXiv preprint arXiv:2503.19470*, 2025.

[4] Z. Cheng, H. Wang, Z. Liu, Y. Guo, Y. Guo, Y. Wang, and H. Wang. Toolspectrum: Towards personalized tool utilization for large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20679–20699, 2025.

[5] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

[6] G. Comanici, E. Bieber, M. Schaekermann, I. Pasupat, N. Sachdeva, I. Dhillon, M. Blistein, O. Ram, D. Zhang, E. Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

[7] Y. Deng, A. Zhou, Y. Yuan, X. Zhang, Y. Zou, and D. Yu. Pe-ma: Parameter-efficient co-evolution of multi-agent systems. *arXiv preprint arXiv:2506.11803*, 2025.

[8] G. Dong, Y. Chen, X. Li, J. Jin, H. Qian, Y. Zhu, H. Mao, G. Zhou, Z. Dou, and J.-R. Wen. Tool-star: Empowering llm-brained multi-tool reasoner via reinforcement learning. *arXiv preprint arXiv:2505.16410*, 2025.

[9] J. Feng, S. Huang, X. Qu, G. Zhang, Y. Qin, B. Zhong, C. Jiang, J. Chi, and W. Zhong. Retool: Reinforcement learning for strategic tool use in llms. *arXiv preprint arXiv:2504.11536*, 2025.

[10] D. Guo, D. Yang, H. Zhang, J. Song, P. Wang, Q. Zhu, R. Xu, R. Zhang, S. Ma, X. Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[11] Z. He, T. Liang, J. Xu, Q. Liu, X. Chen, Y. Wang, L. Song, D. Yu, Z. Liang, W. Wang, et al. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. *arXiv preprint arXiv:2504.11456*, 2025.

[12] X. Ho, A.-K. D. Nguyen, S. Sugawara, and A. Aizawa. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, 2020.

[13] S. Hong, M. Zhuge, J. Chen, X. Zheng, Y. Cheng, J. Wang, C. Zhang, Z. Wang, S. K. S. Yau, Z. Lin, et al. Metagpt: Meta programming for a multi-agent collaborative framework. In *The twelfth international conference on learning representations*, 2023.

[14] J. Hu, Y. Zhang, Q. Han, D. Jiang, X. Zhang, and H.-Y. Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. *arXiv preprint arXiv:2503.24290*, 2025.

[15] C. Huang, S. Gao, Z. Shi, D. Wang, and S. Shang. Ttpa: Token-level tool-use preference alignment training framework with fine-grained evaluation. *arXiv preprint arXiv:2505.20016*, 10, 2025.

[16] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[17] D. Jiang, Y. Lu, Z. Li, Z. Lyu, P. Nie, H. Wang, A. Su, H. Chen, K. Zou, C. Du, et al. Verltool: Towards holistic agentic reinforcement learning with tool use. *arXiv preprint arXiv:2509.01055*, 2025.

[18] G. Jiang, Z. Su, X. Qu, and Y. R. Fung. Xskill: Continual learning from experience and skills in multimodal agents. *arXiv preprint arXiv:2603.12056*, 2026.

[19] B. Jin, H. Zeng, Z. Yue, J. Yoon, S. Arik, D. Wang, H. Zamani, and J. Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

[20] X. Li, G. Dong, J. Jin, Y. Zhang, Y. Zhou, Y. Zhu, P. Zhang, and Z. Dou. Search-o1: Agentic search-enhanced large reasoning models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 5420–5438, 2025.

[21] X. Li, H. Zou, and P. Liu. Torl: Scaling tool-integrated rl. *arXiv preprint arXiv:2503.23383*, 2025.

[22] Z. Li, H. Zhang, S. Han, S. Liu, J. Xie, Y. Zhang, Y. Choi, J. Zou, and P. Lu. In-the-flow agentic system optimization for effective planning and tool use. *arXiv preprint arXiv:2510.05592*, 2025.

[23] J. Liao, M. Wen, J. Wang, and W. Zhang. Marft: Multi-agent reinforcement fine-tuning. *arXiv preprint arXiv:2504.16129*, 2025.

[24] H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever, and K. Cobbe. Let's verify step by step. In *The twelfth international conference on learning representations*, 2023.

[25] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[26] Y. Lin, S. Ye, Y. Yao, Z. Fang, B. Han, and T. Liu. Squidagent: Parallelize wisely, coordinate efficiently. *arXiv preprint arXiv:2610.08647*, 2026.

[27] T. Liu, X. Yu, W. Zhou, J. Gu, and V. Tresp. Focalpo: Enhancing preference optimizing by focusing on correct preference rankings. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 256–267, 2025.

[28] P. Lu, B. Chen, S. Liu, R. Thapa, J. Boen, and J. Zou. Octotools: An agentic framework with extensible tools for complex reasoning. *arXiv preprint arXiv:2502.11271*, 2025.

[29] X. Ma, Q. Liu, D. Jiang, G. Zhang, Z. Ma, and W. Chen. General-reasoner: Advancing llm reasoning across all domains. *arXiv preprint arXiv:2505.14652*, 2025.

[30] MAA. American mathematics competitions. In *American Mathematics Competitions*, 2023.

[31] X. Mai, H. Xu, Z.-Z. Li, W. Wang, J. Hu, Y. Zhang, W. Zhang, et al. Agent rl scaling law: Agent rl with spontaneous code execution for mathematical problem solving. *arXiv preprint arXiv:2505.07773*, 2025.

[32] G. Mialon, C. Fourrier, T. Wolf, Y. LeCun, and T. Scialom. Gaia: a benchmark for general ai assistants. In *The Twelfth International Conference on Learning Representations*, 2023.

[33] S. R. Motwani, C. Smith, R. J. Das, R. Rafailov, I. Laptev, P. H. Torr, F. Pizzati, R. Clark, and C. S. de Witt. Malt: Improving reasoning with multi-agent llm training. *arXiv preprint arXiv:2412.01928*, 2024.

[34] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[35] C. Park, S. Han, X. Guo, A. E. Ozdaglar, K. Zhang, and J.-K. Kim. Maporl: Multi-agent post-co-training for collaborative large language models with reinforcement learning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 30215–30248, 2025.

[36] O. Press, M. Zhang, S. Min, L. Schmidt, N. A. Smith, and M. Lewis. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5687–5711, 2023.

[37] C. Qian, E. C. Acikgoz, Q. He, H. Wang, X. Chen, D. Hakkani-Tür, G. Tur, and H. Ji. Toolrl: Reward is all tool learning needs. *arXiv preprint arXiv:2504.13958*, 2025.

[38] C. Qian, E. C. Acikgoz, H. Wang, X. Chen, A. Sil, D. Hakkani-Tur, G. Tur, and H. Ji. Smart: Self-aware agent for tool overuse mitigation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 4604–4621, 2025.

[39] Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026.

[40] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

[41] D. Rein, B. L. Hou, A. C. Stickland, J. Petty, R. Y. Pang, J. Dirani, J. Michael, and S. R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024.

[42] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[43] Z. Shao, Y. Gong, Y. Shen, M. Huang, N. Duan, and W. Chen. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9248–9274, 2023.

[44] Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. Li, Y. Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[45] H. Song, J. Jiang, Y. Min, J. Chen, Z. Chen, W. X. Zhao, L. Fang, and J.-R. Wen. R1-searcher: Incentivizing the search capability in llms via reinforcement learning. *arXiv preprint arXiv:2503.05592*, 2025.

[46] H. Sun, Z. Qiao, J. Guo, X. Fan, Y. Hou, Y. Jiang, P. Xie, Y. Zhang, F. Huang, and J. Zhou. Zerosearch: Incentivize the search capability of llms without searching. *arXiv preprint arXiv:2505.04588*, 2025.

[47] Q. Team. Qwen2.5: A party of foundation models, September 2024.

[48] H. Trivedi, N. Balasubramanian, T. Khot, and A. Sabharwal. Musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

[49] T.-D. Truong, C. Bobda, N. Agarwal, and K. Luu. Mango: Multimodal attention-based normalizing flow approach to fusion learning. *Advances in Neural Information Processing Systems*, 38:138213–138234, 2026.

[50] T.-D. Truong, H.-Q. Nguyen, X.-B. Nguyen, A. Dowling, X. Li, and K. Luu. Insect-foundation: A foundation model and large multimodal dataset for vision-language insect understanding. *International Journal of Computer Vision*, pages 1–26, 2025.

[51] T.-D. Truong, H.-T. Tran, J. Cothren, B. Raj, and K. Luu. ϕ-dpo: Fairness direct preference optimization approach to continual learning in large multimodal models. *arXiv preprint arXiv:2602.22601*, 2026.

[52] T.-D. Truong, H.-T. Tran, T. Son, B. Raj, and K. Luu. Directed-tokens: A robust multi-modality alignment approach to large language-vision models. *Advances in Neural Information Processing Systems*, 38:140449–140473, 2026.

[53] H. Wang, C. Qian, W. Zhong, X. Chen, J. Qiu, S. Huang, B. Jin, M. Wang, K.-F. Wong, and H. Ji. Acting less is reasoning more! teaching model to act efficiently. *arXiv preprint arXiv:2504.14870*, 2025.

[54] Z. Wang, X. Zheng, K. An, C. Ouyang, J. Cai, Y. Wang, and Y. Wu. Stepsearch: Igniting llms search ability via step-wise proximal policy optimization. *arXiv preprint arXiv:2505.15107*, 2025.

[55] Q. Wu, G. Bansal, J. Zhang, Y. Wu, B. Li, E. Zhu, L. Jiang, X. Zhang, S. Zhang, J. Liu, et al. Autogen: Enabling next-gen llm applications via multi-agent conversations. In *First conference on language modeling*, 2024.

[56] Y. Wu, L. Ma, L. Ding, M. Li, X. Wang, K. Chen, Z. Su, Z. Zhang, C. Huang, Y. Zhang, et al. It takes two: Your grpo is secretly dpo. *arXiv preprint arXiv:2510.00977*, 2025.

[57] Z. Xue, L. Zheng, Q. Liu, Y. Li, X. Zheng, Z. Ma, and B. An. Simpletir: End-to-end reinforcement learning for multi-turn tool-integrated reasoning. *arXiv preprint arXiv:2509.02479*, 2025.

[58] J. Yan, Y. Li, Z. Hu, Z. Wang, G. Cui, X. Qu, Y. Cheng, and Y. Zhang. Learning to reason under off-policy guidance. *arXiv preprint arXiv:2504.14945*, 2025.

[59] A. Yang, B. Zhang, B. Hui, B. Gao, B. Yu, C. Li, D. Liu, J. Tu, J. Zhou, J. Lin, et al. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024.

[60] H. Yang, H. Chen, H. Guo, Y. Chen, C.-S. Lin, S. Hu, J. Hu, X. Wu, and X. Wang. Llm-medqa: Enhancing medical question answering through case studies in large language models. In *2025 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2025.

[61] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. Cohen, R. Salakhutdinov, and C. D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2369–2380, 2018.

[62] Q. Yu, Z. Zhang, R. Zhu, Y. Yuan, X. Zuo, Y. Yue, W. Dai, T. Fan, G. Liu, L. Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

[63] S. Zeng, Q. Wei, W. Brown, O. Frunza, Y. Nevmyvaka, Y. K. Zhao, and M. Hong. Reinforcing multi-turn reasoning in llm agents via turn-level credit assignment. In *ICML 2025 Workshop on Computer Use Agents*, 2025.

[64] W. Zeng, Y. Huang, Q. Liu, W. Liu, K. He, Z. Ma, and J. He. Simplerl-zoo: Investigating and taming zero reinforcement learning for open base models in the wild. *arXiv preprint arXiv:2503.18892*, 2025.

[65] S. Zhang, Y. Dong, J. Zhang, J. Kautz, B. Catanzaro, A. Tao, Q. Wu, Z. Yu, and G. Liu. Nemotron-research-tool-n1: Tool-using language models with reinforced reasoning. *arXiv preprint arXiv:2505.00024*, 2(10), 2025.

[66] D. M. Ziegler, N. Stiennon, J. Wu, T. B. Brown, A. Radford, D. Amodei, P. Christiano, and G. Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

Technical Appendices

A Proof of Theorems

A.1 Proof of Theorem 1

Let q denote a prompt and let $\mathcal{G}(q) = \{\tau_1, \ldots, \tau_K\}, \quad \tau_i \sim \pi_\theta(\cdot|q)$ be a group of $K \geq 2$ sampled trajectories. Assume binary trajectory rewards $r(\tau) \in \{0, 1\}$. Then, we define preferred $\mathcal{P}(q)$ and dispreferred $\mathcal{N}(q)$ trajectory sets, $m = |\mathcal{P}(q)|$, $n = |\mathcal{N}(q)|$, $m + n = K$, and assume non-degenerate groups ($1 \leq m \leq K - 1$). The trajectory-level score function can be defined as $s_\theta(q, \tau) = \nabla_\theta \log \pi_\theta(\tau|q)$.

Theorem 1: Pairwise Equivalence of DPO and K-GRPO up to Positive Weighting. Under unclipped trajectory-level updates with binary rewards, the K-GRPO gradient can be written as

$$g_\theta^{\text{K-GRPO}}(q) = \alpha(q)\, \mathbb{E}_{\tau^+ \sim \text{Unif}(\mathcal{P}(q)),\, \tau^- \sim \text{Unif}(\mathcal{N}(q))} \Big[s_\theta(q, \tau^+) - s_\theta(q, \tau^-) \Big], \tag{14}$$

where the group-dependent positive weight is $\alpha(q) = \sqrt{\hat{p}_q(1 - \hat{p}_q)} = \frac{\sqrt{mn}}{K}$, and $\hat{p}_q = \frac{m}{K}$. Moreover, the DPO gradient can be written as

$$\nabla_\theta L_{\text{DPO}} = -\mathbb{E}_{q, \tau^+, \tau^-} \Big[w_\theta(q, \tau^+, \tau^-) \big(s_\theta(q, \tau^+) - s_\theta(q, \tau^-) \big) \Big], \tag{15}$$

where $w_\theta(q, \tau^+, \tau^-) = \beta \sigma(-\Delta_\theta(q, \tau^+, \tau^-)) > 0$. Therefore K-GRPO and DPO share the same pairwise gradient basis $s_\theta(q, \tau^+) - s_\theta(q, \tau^-)$ and differ only by positive weighting. Hence K-GRPO is pairwise equivalent to DPO up to positive weighting.

Proof. GRPO uses group-normalized advantages [56, 44]. Under binary rewards, all preferred trajectories share one advantage value A^+ and all dispreferred trajectories share another value A^-:

$$A^+ = \sqrt{\frac{1 - \hat{p}_q}{\hat{p}_q}}, \qquad A^- = -\sqrt{\frac{\hat{p}_q}{1 - \hat{p}_q}}, \qquad \hat{p}_q = \frac{m}{K}. \tag{16}$$

Ignoring PPO clipping and working at the trajectory level, the GRPO score-function gradient for one group is

$$g_\theta^{\text{K-GRPO}} = \frac{1}{K} \sum_{i=1}^{K} A_i s_\theta(q, \tau_i). \tag{17}$$

Substituting the binary-reward advantages yields

$$g_\theta^{\text{K-GRPO}} = \frac{1}{K} \left(\sum_{\tau^+ \in \mathcal{P}(q)} A^+ s_\theta(q, \tau^+) + \sum_{\tau^- \in \mathcal{N}(q)} A^- s_\theta(q, \tau^-) \right). \tag{18}$$

Since all preferred trajectories share A^+ and all dispreferred trajectories share A^-,

$$g_\theta^{\text{K-GRPO}} = \frac{1}{K} \left(m A^+ \bar{s}_\theta^+ + n A^- \bar{s}_\theta^- \right), \tag{19}$$

where

$$\bar{s}_\theta^+ = \frac{1}{m} \sum_{\tau^+ \in \mathcal{P}(q)} s_\theta(q, \tau^+), \qquad \bar{s}_\theta^- = \frac{1}{n} \sum_{\tau^- \in \mathcal{N}(q)} s_\theta(q, \tau^-). \tag{20}$$

Substituting the binary advantage values gives

$$mA^+ = \sqrt{mn}, \qquad nA^- = -\sqrt{mn}. \tag{21}$$

Thus

$$g_\theta^{\text{K-GRPO}} = \frac{\sqrt{mn}}{K}\left(\bar{s}_\theta^+ - \bar{s}_\theta^-\right). \tag{22}$$

Observe that the difference between group averages can be written as the expectation over all preferred-dispreferred trajectory pairs:

$$\bar{s}_\theta^+ - \bar{s}_\theta^- = \mathbb{E}_{\tau^+,\tau^-}\Big[s_\theta(q,\tau^+) - s_\theta(q,\tau^-)\Big]. \tag{23}$$

Therefore

$$g_\theta^{\text{K-GRPO}} = \frac{\sqrt{mn}}{K}\,\mathbb{E}_{\tau^+,\tau^-}\Big[s_\theta(q,\tau^+) - s_\theta(q,\tau^-)\Big]. \tag{24}$$

Thus K-GRPO is a positive-weighted average of pairwise preference gradients.

From the DPO objective [56], we can derive the gradient of the DPO loss as follows:

$$\nabla_\theta \mathcal{L}_{\text{DPO}} = -\mathbb{E}\Big[w_\theta(q,\tau^+,\tau^-)\big(s_\theta(q,\tau^+) - s_\theta(q,\tau^-)\big)\Big], \tag{25}$$

with $w_\theta(q,\tau^+,\tau^-) > 0$.

Hence both gradients lie in the cone generated by pairwise preference directions

$$s_\theta(q,\tau^+) - s_\theta(q,\tau^-), \tag{26}$$

differing only by positive weighting.

□

A.2 Proof of Theorem 2

Theorem 2: Suppression of Easy-Majority Dominance in Φ-MPO. Let us define $\mathcal{L}_\gamma(p) = -(1-p)^\gamma \log p$, $p = \sigma(z)$, $\gamma \geq 0$. Then, the margin-gradient magnitude can be formed as $\psi_\gamma(p) := \left|\frac{\partial \mathcal{L}_\gamma(p)}{\partial z}\right|$, Suppose that, at level ℓ, the aligned pairs are partitioned into an easy-majority set $E^{(\ell)}$ and a hard-minority set $H^{(\ell)}$ such that

$$p_{ij}^{(\ell)} \geq 1-\varepsilon \quad \forall (i,j) \in E^{(\ell)}, \qquad p_{ij}^{(\ell)} \leq 1-\delta \quad \forall (i,j) \in H^{(\ell)},$$

for some $0 < \varepsilon < \delta < 1$. Then

$$\sum_{(i,j)\in E^{(\ell)}} \psi_\gamma\left(p_{ij}^{(\ell)}\right) \leq |E^{(\ell)}|(\gamma+1)\varepsilon^{\gamma+1},$$

and

$$\sum_{(i,j)\in H^{(\ell)}} \psi_\gamma\left(p_{ij}^{(\ell)}\right) \geq |H^{(\ell)}|\delta^{\gamma+1}.$$

Therefore,

$$\frac{\sum_{(i,j)\in E^{(\ell)}} \psi_\gamma\left(p_{ij}^{(\ell)}\right)}{\sum_{(i,j)\in H^{(\ell)}} \psi_\gamma\left(p_{ij}^{(\ell)}\right)} \leq \frac{|E^{(\ell)}|}{|H^{(\ell)}|}(\gamma+1)\left(\frac{\varepsilon}{\delta}\right)^{\gamma+1}.$$

Hence Fair MPO suppresses the optimization influence of overrepresented easy pairs and mitigates imbalance caused by easy-majority dominance.

Proof. Since

$$\mathcal{L}_\gamma(p) = -(1-p)^\gamma \log p,$$

we have

$$\frac{\partial \mathcal{L}_\gamma}{\partial z} = (1-p)^\gamma \big(\gamma p \log p + p - 1\big),$$

and thus

$$\psi_\gamma(p) = (1-p)^\gamma \big((1-p) - \gamma p \log p\big).$$

If $p \geq 1-\varepsilon$, then $1-p \leq \varepsilon$ and

$$-\log p \leq \frac{1-p}{p},$$

so

$$-\gamma p \log p \leq \gamma(1-p).$$

Therefore

$$\psi_\gamma(p) \leq (1-p)^\gamma \big((1-p) + \gamma(1-p)\big) = (\gamma+1)(1-p)^{\gamma+1} \leq (\gamma+1)\varepsilon^{\gamma+1}.$$

Summing over $E^{(\ell)}$ yields the first bound.

If $p \leq 1-\delta$, then $1-p \geq \delta$, and since $-\gamma p \log p \geq 0$,

$$\psi_\gamma(p) \geq (1-p)^{\gamma+1} \geq \delta^{\gamma+1}.$$

Summing over $H^{(\ell)}$ yields the second bound. Taking the ratio proves the claim.

B Experiments

B.1 Implementation.

We adopt the implementation of [22] in our experiments. Similar to [22], we adopt Qwen2.5-7B-Instruct for *Action Planner*, *Tool Executor*, *Executive Verifier*, and *Solution Generator* Only the *Action Planner* is trainable, while the others remain fixed during training. Our framework is equipped with five interactive tools: (1) ***Base Generator*** uses Qwen2.5-7B-Instruct, which serves as the default reasoning engine when the planner chooses not to invoke an external tool; (2) ***Python Coder*** generates and executes Python scripts for a given query and returns the execution results; (3) ***Google Search*** retrieves web search results and returns a summary of the top-K retrieved contents; (4) ***Wikipedia Search*** retrieves relevant Wikipedia articles for a given query and returns a summarized response; and (5) ***Web Search*** summarizes the content of a given web page. Following standard protocols [22, 19], we use a learning rate of 1×10^{-6}. The *Action Planner* generates actions with a sampling temperature of 0.5 to balance exploration and exploitation. To stabilize training and prevent policy collapse [19, 22], we set a KL-divergence penalty $\beta = 0.001$. The maximum output length of the planner is set to 2048 tokens to allow sufficient exploration during rollout generation. We use a batch size of 12 and sample $K = 4$ rollouts for each input. It should be noted that if all rollouts obtain the same ranking score $S(\tau)$, we sample an additional K rollouts and recompute the ranking over all $2K$ candidates. If the scores remain tied, we exclude the sample from preference construction for the current update to avoid introducing arbitrary or noisy preference labels. We set the maximum number of turns in each rollout at 3 for training and 10 for evaluation. The reward function is provided by an GPT-4o (LLM-as-a-judge). For stability, the LLM engines used within the tools operate with a temperature of 0.0, ensuring deterministic outputs. The full training process is conducted on 12 NVIDIA L40S GPUs. We adopt the agent prompts and memory update mechanism from [22]. We set the focal hyper-parameter γ to 2.0, and weighting parameters λ_{DPO} and $\lambda_{\Phi-\text{MPO}}$ to 1.0.

B.2 Additional Ablation Study

Effectiveness of Weighting Parameters. Table 7 studies the impact of the weighting parameters λ_{DPO} and $\lambda_{\Phi-\text{MPO}}$, which balance the trajectory-level DPO loss and the Fair MPO in our framework. As shown in Table 7, using equal weights for the two objectives yields the

Table 7: Effectiveness of Weighting Parameters (Accuracy).

λ_{DPO}	$\lambda_{\Phi-\text{MPO}}$	Bamboogle	2Wiki	GAIA	AIME24
0.5	1.0	72.8	73.0	33.1	45.0
1.0	1.0	**81.6**	**81.5**	**41.7**	**50.0**
1.0	0.5	67.2	67.0	26.8	33.3

best performance on all benchmarks, achieving 81.6% on Bamboogle, 81.5% on 2Wiki, 41.7% on GAIA, and 50.0% on AIME24. Reducing the trajectory-level weight to $\lambda_{\mathrm{DPO}} = 0.5$ leads to a moderate drop in performance, showing that the global trajectory-level preference signal remains important for long-horizon reasoning. However, reducing the multi-level weight to $\lambda_{\Phi-\mathrm{MPO}} = 0.5$ causes a much larger degradation across all benchmarks, with the performance dropping to 67.2% on Bamboogle, 67.0% on 2Wiki, 26.8% on GAIA, and 33.3% on AIME24. These results indicate that the fair multi-level objective plays a more critical role in our framework, since it provides fine-grained supervision over intermediate reasoning steps and is essential for effective agentic learning.

Sensitivity to the Number of Rollouts In our approach, we use $K = 4$ as an accuracy-efficiency setting. To examine whether this choice limits performance, we evaluate $K \in \{2, 4, 8\}$ (Table 8). The choice of K represents a clear quality-cost trade-off: larger K increases rollout diversity and the chance of identifying stronger positive-negative pairs, but also increases generation, tool-use, and judge costs. The results show that $K = 4$ provides the best accuracy-efficiency balance. Increasing K from 2 to 4 yields substantial gains of 10.4 percentage points on Bamboogle and 10.0 percentage points on 2Wiki, while increasing K from 4 to 8 provides only marginal gains of 0.8 and 1.0 percentage points, respectively, but increases training time by approximately 40%. Therefore, $K = 4$ is sufficient and offers a strong quality-cost trade-off rather than being an under-justified setting.

Table 8: Sensitivity to the number of rollouts K and the corresponding training cost.

K	**Bamboogle**	**2Wiki**	**Training Hours**
2	71.2%	71.5%	8.0
4	**81.6%**	**81.5%**	**11.4**
8	82.4%	82.5%	16.0

Semantic Reliability of Induced Local Preferences. The induced labels are designed to provide a reliable trajectory-level weak supervision signal, rather than oracle process labels. Our alignment does not pair arbitrary actions: Eqn. 9 matches states using action type, tool choice, memory overlap, and monotonic temporal consistency. This substantially reduces comparisons between unrelated reasoning stages. Eqn. 7 then evaluates each action within its own state and memory context, rather than assuming that one action is universally better. Their reliability is also supported by the consistent empirical gains. Adding MPO to trajectory-level DPO improves all four ablation benchmarks by 13.3–14.9%, while reducing the MPO weight causes a substantially larger degradation than reducing the trajectory-level term. These results show that the aligned local preferences provide a strongly informative signal rather than predominantly noisy supervision. For direct quantification, we additionally conducted a blinded audit of 300 aligned state-action pairs. To avoid outcome and method bias, we evaluated the actions using only the corresponding state and memory context, without access to trajectory outcomes, induced preference labels, or model scores (Table 9). Among comparable pairs, the induced preference agrees with independent local assessment in approximately 88% of cases, while preference reversals occur in only about 10% of all audited pairs. These results directly confirm that the alignment procedure produces predominantly reliable local preference supervision.

Table 9: Semantic Reliability of Induced Local Preferences.

Local Preference Assessment	**Percentage**
Preferred-trajectory action is better	$\sim$72%
Rejected-trajectory action is better	$\sim$10%
Equivalent or incomparable	$\sim$18%
Precision among comparable pairs	$\sim$88%

Sensitivity to Alignment-Set Construction. Our alignment rule is a principled and transparent design that combines action matching, tool matching, and memory overlap to identify semantically comparable reasoning states. The monotonic constraint further preserves temporal consistency and prevents later states from being matched to earlier stages of reasoning. Equal weighting is a natural default because all three similarity components are bounded on the same scale. The substantial improvements from trajectory-level DPO to MPO across all evaluated benchmarks already demonstrate that the aligned state-level supervision is highly effective. To further isolate the contribution of the alignment mechanism, we conduct a focused ablation in Table 10. The proposed alignment consistently achieves the best performance on all four benchmarks. Compared with same-position alignment, it improves accuracy by 10.2–13.3 percentage points; compared with

Table 10: Ablation on alignment-set construction.

Alignment Method	**Bamboogle**	**2Wiki**	**GAIA**	**AIME24**
Proposed monotonic semantic	**81.6**	**81.5**	**41.7**	**50.0**
Same-position alignment	71.2	71.0	31.5	36.7
Random monotonic pairing	69.6	70.0	29.9	36.7

random monotonic pairing, it improves by 11.5–13.3 percentage points. The similar performance of the two weaker baselines further shows that simply adding more state-level pairs is insufficient. The gains come from aligning states that are both semantically related and temporally consistent.

Computational Efficiency Compared with Flow-GRPO and Trajectory DPO. Our computational-efficiency claim is supported both theoretically and empirically. Theoretically, Theorem 1 shows that trajectory-level DPO and unclipped binary-reward K-GRPO share the same preferred-versus-rejected gradient basis, differing only through positive weighting. Therefore, preference optimization preserves the central optimization direction of GRPO without requiring the GRPO objective. As shown in Table 11, Fair-MPO reduces training time by approximately 3.4% relative to Flow-GRPO while improving Bamboogle accuracy by 12.0 percentage points. Moreover, introducing multi-level alignment and focal weighting adds only 0.9 hours over trajectory-level DPO, while improving accuracy by 20.0 percentage points. MPO and Fair-MPO require the same training time, showing that the fairness weighting introduces negligible computational overhead. These results confirm that Fair-MPO achieves a substantially stronger accuracy-efficiency trade-off than both Flow-GRPO and trajectory-level DPO.

Table 11: Accuracy and computational cost comparison.

Method	Bamboogle	Judge Cost	Training (h)
Trajectory DPO	61.6%	∼$4	10.5
Flow-GRPO	69.6%	∼$4	11.8
MPO	76.0%	∼$4	11.4
Fair-MPO	**81.6%**	∼$4	**11.4**

Evaluation Uncertainty and Statistical Reliability. The focal component yields consistent gains across all four benchmarks: +5.6 on Bamboogle, +5.5 on 2Wiki, +5.5 on GAIA, and +6.7 on AIME24. This consistency across search, agentic reasoning, and mathematics shows that the improvement is not benchmark-specific. Table 5 provides further supporting evidence: performance improves steadily as γ increases from 0 to 2, then declines under stronger focal weighting. This directly supports our design: moderate focal weighting reduces easy-pair dominance, while excessive weighting suppresses useful supervision. We additionally report the mean $\pm$ standard deviation over three evaluation runs with different decoding seeds in Table 12. The gains on Bamboogle, 2Wiki, and GAIA are larger than the observed run-to-run variation. AIME24 exhibits higher uncertainty because it contains only 30 evaluation examples. In addition, agentic evaluation involves dynamic external tools, including Search API, whose retrieved results may change over time and introduce environment-level variation beyond decoding randomness. Despite this realistic source of stochasticity, Fair-MPO consistently outperforms MPO across all four benchmarks and exhibits comparable or lower variance.

Table 12: Mean $\pm$ standard deviation over three evaluation runs with different decoding seeds.

Method	Bamboogle	2Wiki	GAIA	AIME24
MPO	76.0 ± 3.1	76.0 ± 3.0	36.2 ± 3.3	43.3 ± 9.1
Φ-MPO	$\mathbf{81.6 \pm 2.5}$	$\mathbf{81.5 \pm 2.7}$	$\mathbf{41.7 \pm 3.4}$	$\mathbf{50.0 \pm 9.0}$

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims in the abstract and introduction are aligned with the proposed method, theoretical analysis, and experimental evaluation. They accurately describe the paper's contributions and scope.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the limitations of the proposed approach in Section 5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The theoretical results and their underlying assumptions are stated in the main paper, with detailed proofs provided in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The details of datasets and implementations are presented in the experimental sections.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [N/A]

Justification: The code will be published may the paper be accepted.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The details of training and testing are presented in the experimental section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [N/A]

   Justification: Following the standard evaluation of agentic learning, we evaluate our model by the standard metrics of the evaluation benchmarks.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The computational resources used in our experiments are presented in the experimental section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The content of the paper and datasets strictly follows the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: The paper does not have a negative societal impact.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not have a risk. The released models will be available may the paper be accepted.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper provides all the references to code, data, and models used in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not introduce the new dataset. The code of the paper will be published may the paper be accepted.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The research in this paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The research in this paper does not involve crowdsourcing nor research with human subjects. Thus, there is no requirement for IRB.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The paper only uses LLMs for grammar correction and word polishing.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.