GHGPO: FINITE-BUDGET GRAPH-HIT CREDIT ASSIGNMENT FOR MULTI-TURN AGENTIC TASKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Fine-grained credit assignment is critical for reinforcement learning in multi-turn agents, yet critic-free group-based methods typically provide only coarse trajectory-level supervision. Recent graph-based approaches improve process credit by using structural distance to success states, but structural efficiency alone does not fully characterize the downstream success probability under the current policy and the finite interaction budget. We propose **Graph-Hit Group Policy Optimization (GHGPO)**, which retains both the state-transition topology and the empirical frequencies of repeated transitions in grouped rollouts to construct an empirical probabilistic state-transition graph reflecting policy-dependent transition behaviors. Based on this graph, GHGPO estimates a finite-budget graph-hit probability to characterize policy-dependent success probability within the available interaction budget, and uses it for fine-grained process credit assignment. Across ALFWorld, WebShop, and Sokoban, GHGPO consistently improves aggregate performance over strong reinforcement learning baselines across different model scales. Ablation studies further validate the importance of finite-budget estimation and the complementarity between policy-dependent success probability and structural efficiency, while GHGPO requires no additional environment interactions and introduces less than 0.2% additional per-iteration computation.

1 INTRODUCTION

Large language models (LLMs) and vision-language models (VLMs) are increasingly used as autonomous agents in tasks that require long-horizon interaction with external environments (Zhou et al., 2024b; Chen et al., 2025), such as embodied control (Driess et al., 2023), web navigation (Zhou et al., 2024a), software operation (Xie et al., 2024), and visual planning (Wu et al., 2025). These tasks require sequences of interdependent decisions that adapt to environmental feedback.

Reinforcement learning (RL) provides a natural framework for improving such agents (Wang et al., 2025a; Zhou et al., 2024b; Feng et al., 2025a). Actor–Critic methods such as PPO (Schulman et al., 2017) can provide token-level credit through learned value functions, but require an additional critic model, incurring extra memory overhead and optimization complexity. In contrast, Group Relative Policy Optimization (GRPO) (Shao et al., 2024) estimates relative advantages from multiple rollout trajectories sampled for the same task without any auxiliary critic, avoiding the overhead of critic training and enabling more scalable RL training. However, GRPO and related group-based methods typically assign the same trajectory-level advantage to all actions within a trajectory, making it difficult to distinguish decisions that advance task progress from ineffective or erroneous steps and thereby diluting action-level learning signals (Li et al., 2026; Xie et al., 2026; Ma et al., 2026). This motivates finer-grained credit assignment for multi-turn agents.

Recent work has begun to exploit multiple interaction trajectories for the same task to achieve finer-grained process credit assignment (Feng et al., 2025b; Cheng et al., 2026). In particular, GraphGPO (Cheng et al., 2026) aggregates grouped rollouts into a state-transition graph and constructs process advantages according to the shortest distance from each state to success. This yields a useful *structural efficiency prior*: actions leading to states closer to success receive higher process credit. Such structural prior provides valuable intermediate supervision for multi-step decision making by revealing efficient routes toward success states. However, *a state closer to success may still have a lower success rate, and states with the same distance to success can exhibit different*

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

success rates, which are empirically demonstrated by our analysis on WebShop (Yao et al., 2022) shown in Sec. 4.1. This motivates us to incorporate the *graph-hit probability*, which is defined as *the probability of a given state transferring to (hitting) the success state in the state-transition graph*, into fine-grained process credit assignment.

To capture such graph-hit probability, as shown in Figure 1, we estimate the state-transition probability matrix of the state-transition graph based on the relative frequencies of outgoing transitions. Based on the state-transition probability matrix, the finite-budget graph-hit probability of reaching the success state from other states within finite interaction steps is derived. Compared with the graph distance, which merely characterizes whether a state is near the success state in the state-transition topology, the proposed finite-budget graph-hit probability measures whether the task can be completed within the remaining interaction budget under the current policy. We further find that the finite-budget constraint is important because a state may eventually reach success with sufficient interaction steps but fail to do so within the remaining budget, and ignoring it substantially weakens the discrimination among actions from the same state. By comprehensively integrating graph distance and finite-budget graph-hit probability, we propose the **Graph-Hit Group Policy Optimization (GHGPO)**. These two signals reflect complementary aspects of process credit, namely structural efficiency and policy-dependent success probability, respectively.

GHGPO consistently improves aggregate performance across diverse multi-turn agent benchmarks and different model scales, while introducing negligible additional computation. Specifically, we evaluate GHGPO on three multi-turn agent benchmarks: ALFWorld (Shridhar et al., 2021), WebShop (Yao et al., 2022), and Sokoban (Schrader, 2018). With Qwen2.5-1.5B-Instruct, GHGPO outperforms GraphGPO by 3.90 percentage points in success rate on ALFWorld, and by 2.98 and 5.98 points in score and success rate on WebShop, respectively. The gains persist at the 7B scale and on vision-based Sokoban. Meanwhile, GHGPO's advantage estimation accounts for only 0.19% and 0.14% of the per-iteration training time on ALFWorld and WebShop, respectively.

Our main contributions are summarized as follows:

- **Identifying policy-dependent success probability as a key dimension of process credit.** We show that structural efficiency alone does not fully characterize success probability under the current policy, and reveal two key findings: (i) a state's success probability must account for the finite interaction budget, as ignoring it substantially weakens action-level discrimination; and (ii) structural distance and finite-budget graph-hit probability capture complementary aspects of process credit, i.e., structural efficiency and policy-dependent success probability, respectively.
- **Policy-dependent finite-budget graph-hit credit assignment.** We propose GHGPO, which retains both the state-transition topology and the empirical frequencies of repeated transitions in grouped rollouts to construct an empirical probabilistic state-transition graph that characterizes the action-selection tendencies of the current policy. Based on it, GHGPO estimates finite-budget graph-hit probabilities to reflect policy-dependent success probability without an auxiliary critic.
- **Extensive empirical validation across agent tasks and model scales.** Experiments on text-based and visual agent benchmarks show that GHGPO achieves consistent aggregate gains across tasks and model scales. Ablation studies validate the effectiveness of finite-budget graph-hit credit and its complementarity with structural distance, while GHGPO requires no additional environment interactions and introduces less than 0.2% additional per-iteration computation.

2 Related Work

RL for LLMs. RL has become a key post-training paradigm for LLMs, including RL from human feedback (RLHF) (Ouyang et al., 2022) and RL with verifiable rewards (RLVR) (Guo et al., 2025). PPO (Schulman et al., 2017) relies on a learned value function for advantage estimation, while critic-free methods such as RLOO (Kool et al., 2019; Ahmadian et al., 2024) and GRPO (Shao et al., 2024) estimate relative advantages from multiple sampled responses without any auxiliary critic. Building on GRPO, subsequent methods further improve group-based policy optimization from different perspectives. Dr. GRPO (Liu et al., 2025) studies objective biases caused by response length and question difficulty, DAPO (Yu et al., 2025) improves sampling and clipping strategies, Clip-Cov (Cui et al., 2025) addresses entropy collapse through covariance-aware masking, and GSPO (Zheng et al., 2025) explores sequence-level importance ratios with length normalization.

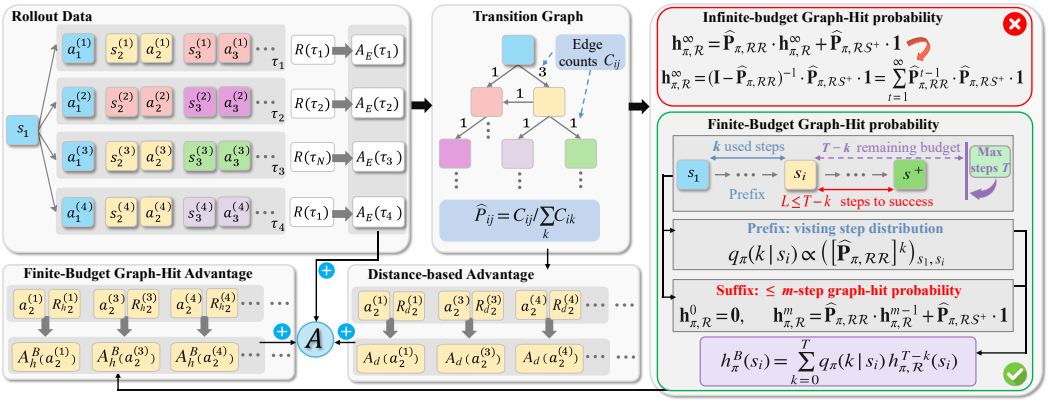


Figure 1: Overview of GHGPO. Grouped rollouts are aggregated into an empirical probabilistic state-transition graph represented by $\widehat{\mathbf{P}}_\pi$ (states with the same color are identical), which preserves state-transition topology and the relative frequencies of outgoing transitions. GHGPO derives two complementary process-credit signals from this graph: distance-based structural efficiency and finite-budget graph-hit probability under the current policy. These signals are combined with trajectory-level outcome credit to form the action-level advantage for policy optimization.

RL for LLM-based Agents. RL has been increasingly applied to LLM-based agents that interact with external environments, including web navigation, application control, search, and other interactive tasks (Zhou et al., 2024b; Putta et al., 2024; Feng et al., 2025a; Chen et al., 2025). Recent studies further extend this paradigm to end-to-end multi-turn training, where policies are optimized over complete interaction trajectories, as in Search-R1 (Jin et al., 2025), RAGEN (Wang et al., 2025b), and WebAgent-R1 (Wei et al., 2025). As interaction horizons grow, another line of work focuses on assigning credit within trajectories at a finer granularity. Turn-PPO (Li et al., 2026) estimates advantages at the turn level with a learned critic, while CAPO (Wang et al., 2026) performs action-level credit assignment by estimating state values at action boundaries. Under critic-free group-based optimization, GiGPO (Feng et al., 2025b) reuses repeated environment states across grouped rollouts to construct step-level comparison groups. GraphGPO (Cheng et al., 2026) further exploits cross-trajectory state-transition structure, aggregating rollouts into a unified graph and deriving process credit from shortest-path distance to success states. Our GHGPO is closely related to this graph-based line, but further incorporates empirical transition frequencies to capture policy-dependent success probability.

3 Preliminaries

Multi-turn Agentic Tasks. Let $x \sim p(\mathcal{X})$ denote a multi-turn interactive task and π_θ an LLM agent parameterized by θ. For each task, the agent interacts with the environment for at most T steps. At step t, the agent observes the current state s_t, samples an action $a_t \sim \pi_\theta(\cdot \mid s_t, x)$, and receives the next state s_{t+1} from the environment. The resulting interaction trajectory is denoted as $\tau = \{(s_t, a_t, s_{t+1})\}_{t=1}^{T}$. In many practical scenarios, the reward $R(\tau) \in \{0, r \mid r > 0\}$ is typically obtained only after the complete trajectory is generated, indicating whether the task is successfully completed within the maximum interaction budget; hence rewards are typically sparse and delayed.

Group-based RL. Group-based RL methods, such as GRPO (Shao et al., 2024), estimate advantages in a critic-free manner, simplifying training. Specifically, for a task x, a group of trajectories $G_x = \{\tau_1, \tau_2, \ldots, \tau_M\}$ is sampled from the LLM π_θ, and each trajectory τ_m receives a scalar reward $R(\tau_m)$. GRPO estimates the trajectory-level advantage as

$$A_E(\tau_m) = \left(R(\tau_m) - \mu\left(\{R(\tau_i)\}_{i=1}^{M}\right)\right) / \sigma\left(\{R(\tau_i)\}_{i=1}^{M}\right), \tag{1}$$

where $\mu(\cdot)$ and $\sigma(\cdot)$ denote the mean and standard deviation. In multi-turn agentic tasks, GRPO assigns the same advantage $A_E(\tau_m)$ to all turns $\{(s_t^m, a_t^m)\}_{t=1}^{T}$ in τ_m. Such coarse-grained credit cannot distinguish effective behaviors from critical errors at the step level, thereby making it difficult for the LLM to learn precise decision-making behaviors.

Graph-Based Step-level Credit Assignment. GraphGPO (Cheng et al., 2026) proposes a graph-based process credit assignment scheme. Given the trajectory group G_x, GraphGPO aggregates its trajectories into a state-transition graph $\mathcal{G} = (\mathcal{S}, \mathcal{E})$. The node set $\mathcal{S}$ contains all observed states, with repeated occurrences of the same state merged into a single node, and each transition $(s_t^m, a_t^m, s_{t+1}^m)$ induces a directed edge in $\mathcal{E}$. Let $d(s)$ denote the minimum cost from state s to a success state:

$$d(s) = \begin{cases} 0, & s \in \mathcal{S}^+, \\ \min\limits_{(s,a,s') \in \mathcal{E}} \left(c(s,a) + d(s') \right), & s \rightsquigarrow \mathcal{S}^+, \\ +\infty, & \text{otherwise}, \end{cases} \tag{2}$$

where $c(s,a)$ denotes the transition cost of executing action a, and $s \rightsquigarrow \mathcal{S}^+$ indicates that there exists a path from s to a success state. Based on $d(s)$, GraphGPO defines the distance-based process reward $R_d(s,a,s') = r\omega^{d(s')+c(s,a)}$, where $r > 0$ is the task-completion reward and $\omega \in (0,1)$ the decay factor. Thus, transitions leading to states closer to success receive higher process rewards.

For each state s, GraphGPO defines a state group by collecting all transitions originating from s:

$$G(s) = \{(s_t^m, a_t^m, s_{t+1}^m) \mid s_t^m = s,\ 1 \le t \le T,\ 1 \le m \le M\}. \tag{3}$$

Within each state group, the distance-based process rewards are obtained by

$$A_d(s,a,s') = (R_d(s,a,s') - \mu_d(G(s)))/\sigma_d(G(s)), \tag{4}$$

where $\mu_d(G(s))$ and $\sigma_d(G(s))$ denote the mean and standard deviation of the distance-based process rewards within $G(s)$, respectively. If a state group contains only one transition instance, the corresponding process advantage is set to 0. In contrast to the trajectory-level advantage A_E, A_d provides state-dependent process credit based on the aggregated transition structure.

4 METHOD

4.1 MOTIVATION: LIMITATIONS OF THE DISTANCE-BASED PROCESS CREDIT

As described in Sec. 3, the distance-based advantage captures structural efficiency in the aggregated stat-transition graph. However, successor states with the same distance to success may have different success probabilities, and a state that is structurally closer to success may have a lower success probability than that of a more distant one. These two discrepancies arise because the success probability depends not only on the graph distance, but also on downstream transition behaviors under the current policy and the remaining interaction budget.

To quantify *the first discrepancy*, we analyze the finite-budget success probability distribution (as estimated in Sec. 4.3) of candidate actions within the same state group during WebShop training (Yao et al., 2022) using distance-based rewards. As shown in the left panel of Figure 2, late in training, 96.1% of state groups contain actions with equal distance (equal-distanced actions) to the task goal, and 31.1% equal-distanced action pairs differ in the finite-budget success probability over 5 percentage points. To show that graph distance and finite-budget success probability induce inconsistent action rankings related to *the second discrepancy*, we calculated the proportion of action pairs that ranked oppositely in terms of distance-based reward and finite-budget success probability. As shown in the right panel of Figure 2, such opposite ranking occurs throughout training, with the rate peaking at 41.7%. Overall, rewarding solely relies on graph distance may not only fail to distinguish equal-distanced actions, but also induce opposite action rankings to those implied by the finite-budget success probability.

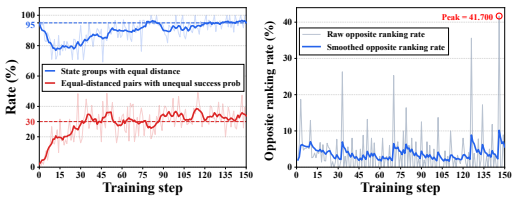


Figure 2: Statistics of distance-based process rewards and finite-budget success probabilities during the training on WebShop. **Left:** Proportion of state groups containing equal-distanced actions, and proportion of equal-distanced action pairs with over 5% differences in finite-budget success probability. **Right:** Rate of opposite action rankings between distance-based rewards and finite-budget success probabilities.

These observations motivate us to capture not only the graph distance but also the finite-budget success probability between each state to the task goal to assign step-level rewards.

4.2 State-Transition Matrix Induced by Policy and Environment

Given a task x, denote $\mathcal{T}(s' \mid s, a, x)$ as the probability that the environment transitions to state s' after executing action a at state s. Then, the one-step transition probability distribution jointly induced by the current policy π_θ and the environment is

$$P_\pi(s' \mid s, x) = \sum_a \pi_\theta(a \mid s, x) \cdot \mathcal{T}(s' \mid s, a, x), \tag{5}$$

which reflects both the action distribution of the current policy and the corresponding environment dynamics. P_π changes as the policy updates. For simplicity, we omit its dependence on x below when no ambiguity arises.

In practice, the exact P_π is typically difficult to obtain. We therefore estimate it empirically from the grouped rollouts G_x generated by the current policy. Let $\mathcal{S} = \{s_1, s_2, \ldots, s_N\}$ denote the set of states observed in these trajectories. For any two states s_i and s_j, we count the number of transitions from s_i to s_j observed in the grouped rollouts:

$$C_{ij} = \sum_{m=1}^{M} \sum_{t=1}^{T} \mathbf{1}\left[s_t^m = s_i,\, s_{t+1}^m = s_j\right], \tag{6}$$

which preserves the empirical frequency of repeated state transitions. Use the number of observed outgoing transitions from state s_i, i.e., $N_i = \sum_{k=1}^{N} C_{ik}$, to normalize these empirical counts:

$$\widehat{P}_{\pi,ij} = \frac{C_{ij}}{N_i}, \quad N_i > 0, \qquad \widehat{P}_{\pi,ij} = 0, \quad N_i = 0. \tag{7}$$

This yields the empirical transition probability matrix $\widehat{\mathbf{P}}_\pi = (\widehat{P}_{\pi,ij})_{N\times N}$ of the state-transition graph $\mathcal{G}$, whose entries characterize the relative one-step transition tendencies between states under the current policy.

To estimate the probability of reaching success from each state, we partition the observed states into three sets: the success states $\mathcal{S}^+$, the graph-unreachable states $\mathcal{S}^- = \{s \in \mathcal{S} \mid d(s) = +\infty\}$, and the remaining intermediate states $\mathcal{R}$. Here, $d(s)$ is defined in Eq. (2). After reordering the states according to the partition $(\mathcal{R}, \mathcal{S}^-, \mathcal{S}^+)$, $\widehat{\mathbf{P}}_\pi$ can be written in block matrix form as follows:

$$\widehat{\mathbf{P}}_\pi = \begin{bmatrix} \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{R}} & \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{S}^-} & \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{S}^+} \\ \mathbf{0} & \widehat{\mathbf{P}}_{\pi,\mathcal{S}^-\mathcal{S}^-} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}. \tag{8}$$

Here, $\mathbf{0}$ is the zero matrix, and $\widehat{\mathbf{P}}_{\pi,\mathcal{A}\mathcal{B}}$ denotes the transition sub-matrix from state set $\mathcal{A}$ to state set $\mathcal{B}$. Since states in $\mathcal{S}^-$ cannot subsequently reach success and success states terminate the task, the subsequent computation of state success probabilities only requires $\widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{R}}$ and $\widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{S}^+}$.

4.3 Estimation of Finite-Budget Graph-Hit Probability and Hit-Based Advantage

Estimation of Finite-Budget Graph-Hit Probability. Based on the estimated state-transition matrix $\widehat{\mathbf{P}}_\pi$, we calculate the graph-hit probability for each state and use it as an empirical proxy for policy-dependent success probability. Since $h_\pi(s) = 1, \forall s \in \mathcal{S}^+$, and $h_\pi(s) = 0, \forall s \in \mathcal{S}^-$, we mainly focus on the estimation of success probabilities of the intermediate states in $\mathcal{R}$.

In real-world applications, the interaction budget is finite, which is a constraint that cannot be ignored when estimating the success probability of states. Denote the vector of probabilities of intermediate states reaching the success state within at most m steps as $\mathbf{h}_{\pi,\mathcal{R}}^m \in [0,1]^{|\mathcal{R}|}$, according to the one-step state transition relationship, we have

$$\mathbf{h}_{\pi,\mathcal{R}}^0 = \mathbf{0}, \qquad \mathbf{h}_{\pi,\mathcal{R}}^m = \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{R}} \cdot \mathbf{h}_{\pi,\mathcal{R}}^{m-1} + \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{S}^+} \cdot \mathbf{1}, \quad m \geq 1, \tag{9}$$

Meanwhile, the same state may be visited at different interaction steps, so the remaining budget also affects its success probability. We therefore marginalize the step-dependent success probability over the visiting step distribution under the current policy and derive the *finite-budget graph-hit probability* of each state. Proposition 1 gives the resultant state-level finite-budget graph-hit probability, with its proof being provided in Appendix B.

Proposition 1 (Finite-budget graph-hit probability)**.** *Let the initial state be s_1 and the maximum number of interaction steps be T. For any intermediate state $s_i \in \mathcal{R}$, its normalized visiting step distribution is*

$$q_\pi(k \mid s_i) = \frac{\mu_k(s_i)}{\sum_{\ell=0}^{T} \mu_\ell(s_i)}, \qquad \mu_k(s_i) = \left(\left[\widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{R}} \right]^k \right)_{s_1, s_i}, \qquad k = 0, \ldots, T, \tag{10}$$

where $\mu_k(s_i)$ denotes the probability of visiting state s_i at step k in the empirical transition process defined by $\widehat{\mathbf{P}}_\pi$. Then, applying the law of total probability over the partition induced by the visiting step, the finite-budget graph-hit probability of state s_i ($h_\pi^B(s_i)$) is calculated by marginalizing over its visiting step distribution $K \sim q_\pi(\cdot \mid s_i)$:

$$h_\pi^B(s_i) = \mathbb{E}_{K \sim q_\pi(\cdot \mid s_i)} \left[h_{\pi,\mathcal{R}}^{T-K}(s_i) \right] = \frac{\sum_{k=0}^{T} \mu_k(s_i)\, h_{\pi,\mathcal{R}}^{T-k}(s_i)}{\sum_{k=0}^{T} \mu_k(s_i)}. \tag{11}$$

The visiting step distribution is derived from the aggregated empirical probabilistic state-transition graph rather than trajectory-level visitation positions. This reduces reliance on sparse occurrence-level statistics and allows the estimation of finite-budget graph-hit probabilities to reuse transition evidence aggregated across grouped rollouts. This design choice is further supported by the empirical comparison in Appendix D.1. Thus, $h_\pi^B(s_i)$ not only considers the remaining interaction budget but also captures the policy-dependent transition behavior.

Notably, when the interaction budget is infinite, i.e., $m \to \infty$, it holds that

$$\mathbf{h}_{\pi,\mathcal{R}}^{\infty} = \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{R}} \cdot \mathbf{h}_{\pi,\mathcal{R}}^{\infty} + \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{S}^+} \cdot \mathbf{1}, \tag{12}$$

where $\mathbf{h}_{\pi,\mathcal{R}}^{\infty} \in [0,1]^{|\mathcal{R}|}$ denote the vector of infinite-budget graph-hit probability of intermediate states. When $\mathbf{I} - \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{R}}$ is invertible, $\mathbf{h}_{\pi,\mathcal{R}}^{\infty} = (\mathbf{I} - \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{R}})^{-1} \cdot \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{S}^+} \cdot \mathbf{1}$. Further, if the spectral radius of $\widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{R}}$ is less than 1, we have $\mathbf{h}_{\pi,\mathcal{R}}^{\infty} = \sum_{t=1}^{\infty} \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{R}}^{\,t-1} \cdot \widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{S}^+} \cdot \mathbf{1}$.

Hit-Based Process Advantage. For a state s, its state group $G(s)$ is defined in Eq. (3). The vector of $\{h_\pi^B(s_i)\}_{i=1}^N$ is denoted as $\mathbf{h}_\pi^B$. Given a graph-hit probability estimate $\mathbf{h}_\pi \in \{\mathbf{h}_\pi^\infty, \mathbf{h}_\pi^B\}$, we use the graph-hit probability of the successor state reached by an action as the process reward:

$$R_h(s_i, a_i, s_j) = h_\pi(s_j). \tag{13}$$

Thus, actions leading to successor states with higher estimated success probabilities receive higher process rewards. We then normalize this process reward within the state group $G(s)$ to obtain

$$A_h(s_i, a_i, s_j) = \left(R_h(s_i, a_i, s_j) - \mu_h(G(s))\right) / \sigma_h(G(s)), \tag{14}$$

where $\mu_h(G(s))$ and $\sigma_h(G(s))$ denote the mean and standard deviation of the hit-based process rewards within state group $G(s)$, respectively. We use A_h^B (A_h^∞) to denote the resultant process advantage when $\mathbf{h}_\pi^B$ ($\mathbf{h}_\pi^\infty$) is used to define the reward.

Limitations of Infinite-Budget Credit. Compared with $\mathbf{h}_\pi^B$, $\mathbf{h}_\pi^\infty$ ignores the finite interaction budget constraint and the resultant credit assignment may become blind to how fast success is reached, as detoured states and fast-to-success states may receive identical advantage scores, which systematically degrades the discriminative power of credit assignment. We verify the above analysis on WebShop by collecting the within-group standard-deviation ratio of $\mathbf{h}_\pi^\infty$ to $\mathbf{h}_\pi^B$ and the sign distribution of A_h^∞ for transitions with positive or negative A_h^B. We further validate this design choice through an ablation study. Refer to Appendix D.2 for more details.

4.4 Graph-Hit Group Policy Optimization

GHGPO integrates three complementary sources of credit for multi-turn agent RL: the trajectory-level advantage A_E reflecting the overall outcome of the trajectory, the graph distance based advantage A_d capturing the state-transition graph's structural efficiency to success, the finite-budget graph-hit probability based advantage A_h^B capturing policy-dependent success probability.

Table 1: The test performance on ALFWorld and WebShop. We report success rates (%) for the six ALFWorld task types and the overall success rate (All), and task score and success rate (Succ.) for WebShop. RL results are reported as mean and standard deviation over three random seeds. The best performances are highlighted in bold.

Type	Method	ALFWorld							WebShop	
		Pick	Clean	Cool	Look	Heat	Pick2	All	Score	Succ.
Closed-Source Models										
Prompting	GPT-4o	75.3	60.8	31.2	56.7	21.6	49.8	48.0	31.8	23.7
Prompting	Gemini-2.5-Pro	92.8	63.3	62.1	69.0	26.6	58.7	60.3	42.5	35.9
Qwen2.5-1.5B-Instruct										
Prompting	Qwen2.5	5.9	5.5	3.3	9.7	4.2	0.0	4.1	23.1	5.2
Prompting	ReAct	17.4	20.5	15.7	6.2	7.7	2.0	12.8	40.1	11.3
Prompting	Reflexion	35.3	22.2	21.7	13.6	19.4	3.7	21.8	55.8	21.9
RL Training	PPO	$64.8_{(3.5)}$	$40.5_{(6.9)}$	$57.1_{(4.9)}$	$60.6_{(6.6)}$	$46.4_{(4.0)}$	$47.4_{(1.9)}$	$54.4_{(3.1)}$	$73.8_{(3.0)}$	$51.5_{(2.9)}$
RL Training	RLOO	$88.3_{(3.0)}$	$52.8_{(8.6)}$	$71.0_{(5.9)}$	$62.8_{(8.7)}$	$66.4_{(5.5)}$	$56.9_{(4.7)}$	$69.7_{(2.5)}$	$73.9_{(5.6)}$	$52.1_{(6.7)}$
RL Training	GRPO	$82.89_{(3.62)}$	$82.14_{(6.37)}$	$73.86_{(6.84)}$	$78.57_{(0.00)}$	$77.78_{(4.54)}$	$71.43_{(3.89)}$	$77.86_{(1.33)}$	$84.73_{(0.49)}$	$71.35_{(2.05)}$
RL Training	GiGPO	$\mathbf{98.81}_{(1.68)}$	$95.16_{(3.89)}$	$81.40_{(5.60)}$	$78.57_{(0.00)}$	$94.44_{(0.00)}$	$93.65_{(5.94)}$	$90.88_{(0.97)}$	$87.94_{(0.43)}$	$73.83_{(2.30)}$
RL Training	GraphGPO	$95.15_{(1.62)}$	$\mathbf{100.0}_{(0.00)}$	$85.26_{(2.58)}$	$85.71_{(5.83)}$	$96.30_{(2.61)}$	$93.65_{(2.24)}$	$92.71_{(1.32)}$	$89.29_{(1.48)}$	$78.65_{(3.86)}$
RL Training	**GHGPO**	$93.61_{(0.12)}$	$\mathbf{100.0}_{(0.00)}$	$\mathbf{95.83}_{(7.22)}$	$\mathbf{97.22}_{(4.82)}$	$\mathbf{98.15}_{(3.21)}$	$\mathbf{97.10}_{(5.02)}$	$\mathbf{96.61}_{(1.96)}$	$\mathbf{92.27}_{(2.46)}$	$\mathbf{84.63}_{(2.74)}$
Qwen2.5-7B-Instruct										
Prompting	Qwen2.5	33.4	21.6	19.3	6.9	2.8	3.2	14.8	26.4	7.8
Prompting	ReAct	48.5	35.4	34.3	13.2	18.2	17.6	31.2	46.2	19.5
Prompting	Reflexion	62.0	41.6	44.9	30.9	36.3	23.8	42.7	58.1	28.8
RL Training	PPO	$92.3_{(4.0)}$	$64.0_{(8.4)}$	$92.5_{(2.4)}$	$89.5_{(7.0)}$	$80.3_{(2.0)}$	$68.8_{(8.3)}$	$80.4_{(2.7)}$	$81.4_{(3.1)}$	$68.7_{(5.1)}$
RL Training	RLOO	$87.6_{(4.3)}$	$78.2_{(8.3)}$	$87.3_{(5.8)}$	$81.3_{(7.6)}$	$71.9_{(5.2)}$	$48.9_{(8.4)}$	$75.5_{(4.6)}$	$80.3_{(3.2)}$	$65.7_{(4.0)}$
RL Training	GRPO	$88.98_{(5.30)}$	$91.98_{(4.43)}$	$77.89_{(4.58)}$	$78.57_{(0.00)}$	$90.74_{(5.24)}$	$71.43_{(3.89)}$	$83.33_{(2.05)}$	$84.31_{(1.27)}$	$75.00_{(2.78)}$
RL Training	GiGPO	$97.53_{(1.75)}$	$\mathbf{100.0}_{(0.00)}$	$83.98_{(1.32)}$	$90.48_{(6.73)}$	$94.44_{(0.00)}$	$\mathbf{100.0}_{(0.00)}$	$94.27_{(1.33)}$	$86.72_{(1.44)}$	$78.38_{(1.94)}$
RL Training	GraphGPO	$\mathbf{100.0}_{(0.00)}$	$\mathbf{100.0}_{(0.00)}$	$91.40_{(1.50)}$	$92.86_{(5.83)}$	$94.44_{(0.00)}$	$92.06_{(2.24)}$	$95.31_{(1.10)}$	$86.94_{(0.68)}$	$80.31_{(1.33)}$
RL Training	**GHGPO**	$\mathbf{100.00}_{(0.00)}$	$\mathbf{100.00}_{(0.00)}$	$\mathbf{100.00}_{(0.00)}$	$\mathbf{96.97}_{(5.25)}$	$\mathbf{100.00}_{(0.00)}$	$93.04_{(3.60)}$	$\mathbf{98.69}_{(0.90)}$	$\mathbf{92.11}_{(1.26)}$	$\mathbf{84.89}_{(1.19)}$

We therefore define the final action-level advantage for each action transition $(s_t, a_t, s_{t+1}) \in \tau$ as

$$A(s_t, a_t, s_{t+1}) = \lambda_E A_E(\tau) + \lambda_h A_h^B(s_t, a_t, s_{t+1}) + \lambda_d A_d(s_t, a_t, s_{t+1}), \tag{15}$$

where λ_E, λ_h, and λ_d control the weights of the three advantage signals, respectively.

During optimization, we broadcast the action-level advantage A to the valid response tokens corresponding to the action at the current interaction step and adopt the clipped policy objective:

$$\mathcal{J}(\theta) = \mathbb{E}\left[\min\left(r_t(\theta)A(s_t, a_t, s_{t+1}), \text{clip}\left(r_t(\theta), 1-\epsilon, 1+\epsilon\right)A(s_t, a_t, s_{t+1})\right) - \beta D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})\right], \tag{16}$$

where $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$. The pseudo-code is shown in Algorithm 1 of Appendix A.

5 Experiments

In this section, we conduct systematic experiments to evaluate the effectiveness of GHGPO.

5.1 Experimental Setup

Benchmarks. We evaluate GHGPO across three multi-turn agentic benchmarks covering embodied interaction, web navigation, and visual planning: ALFWorld (Shridhar et al., 2021), WebShop (Yao et al., 2022), and Sokoban (Schrader, 2018). We report the success rate on all benchmarks and additionally report task scores on WebShop. Benchmark details are provided in Appendix C.1.

Baselines. For ALFWorld and WebShop, we compare GHGPO with three categories of baselines: (1) closed-source models: GPT-4o (OpenAI, 2024) and Gemini-2.5-Pro (Gemini Team, Google, 2023); (2) prompting-based methods: base-model prompting, ReAct (Yao et al., 2023), and Reflexion (Shinn et al., 2023); and (3) RL methods: PPO (Schulman et al., 2017), RLOO (Kool et al., 2019; Ahmadian et al., 2024), GRPO (Shao et al., 2024), GiGPO (Feng et al., 2025b), and GraphGPO (Cheng et al., 2026). For Sokoban, we compare against the base VLM and group-based RL methods: GRPO, GiGPO, and GraphGPO. Baseline details are provided in Appendix C.2.

Implementation Details. Following the prior work (Cheng et al., 2026), we use Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct (Qwen, 2025) for ALFWorld and WebShop, and Qwen2.5-VL-3B-Instruct (Bai et al., 2025) for Sokoban. The agent retains the two most recent interaction steps as

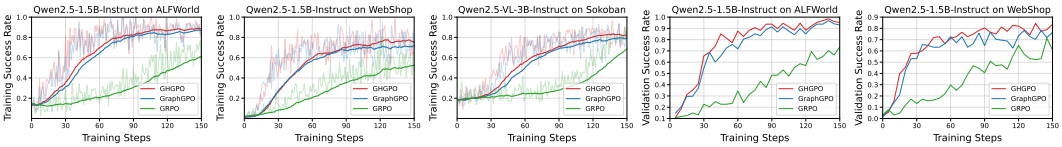


Figure 3: Training and validation dynamics of GRPO, GraphGPO, and GHGPO on ALFWorld, WebShop, and Sokoban. Light curves show the raw success rates, while dark curves show their exponential moving averages (EMA) with $\alpha = 0.95$. Sokoban has no fixed validation split because its environments are randomly generated.

short-term memory and generates its reasoning and actions within `<think>...</think>` and `<action>...</action>` tags, respectively. For fair comparison, GHGPO follows the same benchmark-specific base models, environment configurations, and rollout budgets as the compared RL methods. The rollout group size is 8 for all group-based methods. For GHGPO, we set $\lambda_E = \lambda_d = \lambda_h = 1$ without benchmark- or model-specific tuning. Unless otherwise specified, we report mean and standard deviation over three random seeds. Full details are in Appendix C.3.

5.2 Experimental Results

Main Results. As shown in Tables 1-2, GHGPO consistently achieves the best aggregate performance among the compared RL methods. With Qwen2.5-1.5B-Instruct, GHGPO outperforms GraphGPO by 3.90 percentage points in overall ALFWorld success rate, and by 2.98 and 5.98 points in WebShop task score and success rate, respectively. The gains persist with Qwen2.5-7B-Instruct, where the improvements are 3.38, 5.17, and 4.58 points, respectively. These gains suggest that incorporating policy-dependent finite-budget graph-hit probability provides complementary information beyond structural distance, and that its benefit persists across model scales. On the vision-based Sokoban benchmark, GHGPO further improves the success rate from 86.98% to 88.28% over GraphGPO, suggesting that the proposed credit mechanism also generalizes to VLM-based long-horizon planning.

Table 2: The test performance on Sokoban with Qwen2.5-VL-3B-Instruct. We report the success rate (%) on the 6×6 setting. The best performance is highlighted in bold.

Type	Method	Sokoban [6×6]
Prompting	Qwen2.5-VL	11.7
RL Training	GRPO	67.1
RL Training	GiGPO	76.92
RL Training	GraphGPO	$86.98_{(0.73)}$
RL Training	**GHGPO**	$\mathbf{88.28}_{(1.35)}$

Ablation Studies. Table 3 evaluates two key design choices of GHGPO: the complementary roles of the structural distance advantage A_d and the finite-budget graph-hit advantage A_h^B. Removing either A_d or A_h^B degrades aggregate performance across ALFWorld, WebShop, and Sokoban, indicating that both process-level signals contribute to the final performance. Notably, the variant without A_d still outperforms GraphGPO on all aggregate metrics, showing that the finite-budget graph-hit credit remains effective even without structural distance information. Meanwhile, the full GHGPO consistently performs best, supporting the complementarity between the two process-level signals: A_d captures structural efficiency, whereas A_h^B captures policy-dependent finite-budget success probability. We observe the same component-ablation trend with Qwen2.5-7B-Instruct on ALFWorld; the corresponding results are reported in Appendix D.3.

Table 3: Ablation study on ALFWorld, WebShop, and Sokoban. ALFWorld and WebShop use Qwen2.5-1.5B-Instruct, while Sokoban uses Qwen2.5-VL-3B-Instruct. Removing A_h^B reduces GHGPO to GraphGPO under the same experimental configuration.

Method	ALFWorld							WebShop		Sokoban
	Pick	Clean	Cool	Look	Heat	Pick2	All	Score	Succ.	Succ.
GHGPO	$93.61_{(0.12)}$	$\mathbf{100.0}_{(0.00)}$	$\mathbf{95.83}_{(7.22)}$	$\mathbf{97.22}_{(4.82)}$	$98.15_{(3.21)}$	$\mathbf{97.10}_{(5.02)}$	$\mathbf{96.61}_{(1.96)}$	$\mathbf{92.27}_{(2.46)}$	$\mathbf{84.63}_{(2.74)}$	$\mathbf{88.28}_{(1.35)}$
$-A_d$	$94.79_{(6.81)}$	$98.72_{(2.22)}$	$87.38_{(3.99)}$	$96.67_{(5.77)}$	$\mathbf{100.0}_{(0.00)}$	$88.26_{(6.42)}$	$93.75_{(3.58)}$	$91.19_{(1.85)}$	$83.33_{(1.62)}$	$87.24_{(2.74)}$
$-A_h^B$	$95.15_{(1.62)}$	$\mathbf{100.0}_{(0.00)}$	$85.26_{(2.58)}$	$85.71_{(5.83)}$	$96.30_{(2.61)}$	$93.65_{(2.24)}$	$92.71_{(1.32)}$	$89.29_{(1.48)}$	$78.65_{(3.86)}$	$86.98_{(0.73)}$

Training Dynamics. To examine the learning dynamics of different credit-assignment methods, we report the training and validation success rates of GRPO, GraphGPO, and GHGPO in Figure 3. For Sokoban, we report only training performance because its environments are randomly generated without a fixed validation split. Across three benchmarks, GraphGPO and GHGPO generally maintain higher success rates during training than trajectory-level GRPO, highlighting the benefit of fine-grained process credit for multi-turn agent training. GHGPO generally matches or outperforms GraphGPO throughout training, with clearer gains on WebShop and Sokoban during the middle and later stages. This pattern suggests that policy-dependent finite-budget graph-hit probability provides complementary action-level information beyond structural distance, helping distinguish actions by their likelihood of success within the remaining interaction budget. Appendix D.4 further analyzes the evolution of the empirical transition structure during training. As training progresses, states and transition edges receive increasingly repeated observations, while the average state out-degree gradually decreases, indicating that rollout behavior becomes more concentrated.

Computational Overhead. GHGPO introduces additional computation only during advantage estimation. Let e denote the number of nonzero entries in $\widehat{\mathbf{P}}_{\pi,\mathcal{RR}}$, i.e., the number of distinct directed edges between intermediate states in the empirical transition graph. The finite-budget graph-hit probabilities and visiting step probabilities are computed through T rounds of dynamic programming, over the sparse empirical transition graph. The time complexity is $O(Te)$. To quantify the overhead, Figure 4 reports the per-iteration training time breakdown on ALFWorld and WebShop. In both environments, rollout and policy update account for the majority of the training time, whereas the additional advantage computation introduced by GHGPO accounts for only 0.19% and 0.14% of the total runtime on ALFWorld and WebShop, respectively. Therefore, GHGPO provides policy-dependent finite-budget process credit with negligible computational overhead, while requiring neither an auxiliary critic nor additional environment interactions.

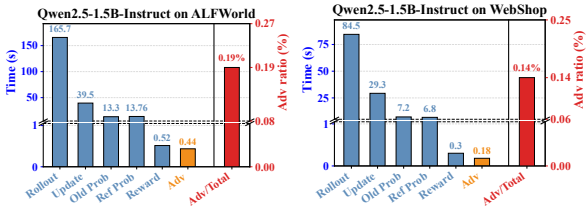


Figure 4: Per-iteration training-time breakdown of GHGPO on ALFWorld (left) and WebShop (right). Blue bars show training stages shared with group-based RL methods, while orange bars denote GHGPO-specific advantage computation. Broken y-axes are used to make the small additional cost visible.

Additional results are provided in the appendices. Appendix D includes effect of marginalization over the visiting step distribution, effect of ignoring the remaining interaction budget, additional ablation study on Qwen2.5-7B-Instruct, an analysis of the evolution of the empirical transition graph during training, and rollout group-size analysis, while Appendix E presents qualitative case studies.

6 Conclusion

This work identifies policy-dependent finite-budget success probability as a key dimension of fine-grained process credit for RL in multi-turn agents. To model this dimension, we propose Graph-Hit Group Policy Optimization (GHGPO), which constructs an empirical state-transition graph from grouped rollouts by preserving both transition topology and repeated transition frequencies. Based on this graph, GHGPO estimates finite-budget graph-hit probabilities to characterize policy-dependent success probability under the remaining interaction budget, and combines this signal with trajectory-level outcome credit and distance-based structural credit for action-level optimization. Experiments on ALFWorld, WebShop, and Sokoban across different model scales show consistent performance improvements, while ablation studies validate both the importance of finite-budget estimation and the complementary roles of structural efficiency and policy-dependent success probability. GHGPO also introduces negligible computational overhead and requires neither an auxiliary Critic nor additional environment interactions. A promising future direction is to explore more robust state-matching and aggregation strategies, such as semantic or task-structure-aware equivalence, to share transition evidence across behaviorally similar states.

AI USE STATEMENT

Generative AI tools were used only for language polishing and checking compliance with the submission template. They were not used to generate research ideas, develop the proposed method, conduct experiments, or analyze experimental results. All AI-assisted content was reviewed by the authors, who take full responsibility for the final content of this work.

ETHICS STATEMENT

This work studies reinforcement learning methods for multi-turn agents in standard benchmark environments and does not involve human subjects, personally identifiable information, or real-world deployment. The proposed method is intended to improve credit assignment and training efficiency in agentic reinforcement learning. While such advances may contribute to more capable autonomous agents, deployment in real-world environments may introduce risks associated with erroneous or unintended actions. Appropriate safeguards, monitoring, and human oversight should therefore be considered when applying these techniques beyond controlled benchmark settings.

REPRODUCIBILITY STATEMENT

Our implementation is built on the `verl-agent` framework (Feng et al., 2025b). We provide the pseudocode of GHGPO and the proof of Proposition 1 in Appendices A and B, respectively. Appendix C provides benchmark and baseline descriptions, training and implementation details, prompt templates, and computing configurations. Additional ablations and analyses are reported in Appendix D. Together, these materials provide the algorithmic and experimental details needed to reproduce the reported results. The source code will be released upon acceptance.

REFERENCES

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

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Kevin Chen, Marco Cusumano-Towner, Brody Huval, Aleksei Petrenko, Jackson Hamburger, Vladlen Koltun, and Philipp Krähenbühl. Reinforcement learning for long-horizon interactive LLM agents. *arXiv preprint arXiv:2502.01600*, 2025.

Xin Cheng, Shuo He, Lang Feng, Haiyang Xu, Ming Yan, Lei Feng, and Bo An. Beyond trajectory-level attribution: Graph-based credit assignment for agentic reinforcement learning. In *ICML*, 2026.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, Zhiyuan Liu, Hao Peng, Lei Bai, Wanli Ouyang, Yu Cheng, Bowen Zhou, and Ning Ding. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. PaLM-E: An embodied multimodal language model. In *ICML*, volume 202, pp. 8469–8488, 2023.

Lang Feng, Weihao Tan, Zhiyi Lyu, Longtao Zheng, Haiyang Xu, Ming Yan, Fei Huang, and Bo An. Towards efficient online tuning of VLM agents via counterfactual soft reinforcement learning. In *ICML*, volume 267, pp. 16884–16903, 2025a.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. Group-in-group policy optimization for llm agent training. In *NeurIPS*, volume 38, pp. 46375–46408, 2025b.

Gemini Team, Google. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan O. Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-R1: Training LLMs to reason and leverage search engines with reinforcement learning. In *COLM*, 2025.

Wouter Kool, Herke van Hoof, and Max Welling. Buy 4 REINFORCE samples, get a baseline for free! In *ICLR 2019 Workshop*, 2019.

Junbo Li, Peng Zhou, Rui Meng, Meet P. Vadera, Lihong Li, and Yang Li. Turn-PPO: Turn-level advantage estimation with PPO for improved multi-turn RL in agentic LLMs. In *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 6227–6243, 2026.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. In *COLM*, 2025.

Da Ma, Ziyue Yang, Hongshen Xu, Haotian Fang, Kai Yu, and Lu Chen. Empowering LLM tool invocation with tool-call reward model. In *ICLR*, pp. 34322–34352, 2026.

OpenAI. GPT-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *NeurIPS*, volume 35, pp. 27730–27744, 2022.

Pranav Putta, Edmund Mills, Naman Garg, Sumeet Motwani, Chelsea Finn, Divyansh Garg, and Rafael Rafailov. Agent Q: Advanced reasoning and learning for autonomous AI agents. *arXiv preprint arXiv:2408.07199*, 2024.

Qwen. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025.

Max-Philipp B. Schrader. gym-sokoban. `https://github.com/mpSchrader/gym-sokoban`, 2018.

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

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

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *NeurIPS*, volume 36, pp. 8634–8652, 2023.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. Alfworld: Aligning text and embodied environments for interactive learning. In *ICLR*, 2021.

Daoyu Wang, Qingchuan Li, Mingyue Cheng, Jie Ouyang, Shuo Yu, Chunli Liu, Shijin Wang, Qi Liu, and Enhong Chen. CAPO: Critic-guided action-aligned policy optimization for advancing LLM agent capabilities. *arXiv preprint arXiv:2604.18401*, 2026.

Kangrui Wang, Pingyue Zhang, Zihan Wang, Yaning Gao, Linjie Li, Qineng Wang, Hanyang Chen, Yiping Lu, Zhengyuan Yang, Lijuan Wang, Ranjay Krishna, Jiajun Wu, Fei-Fei Li, Yejin Choi, and Manling Li. VAGEN: Reinforcing world model reasoning for multi-turn VLM agents. In *NeurIPS*, volume 38, pp. 172871–172933, 2025a.

Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, Eli Gottlieb, Yiping Lu, Kyunghyun Cho, Jiajun Wu, Fei-Fei Li, Lijuan Wang, Yejin Choi, and Manling Li. RAGEN: Understanding self-evolution in LLM agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*, 2025b.

Zhepei Wei, Wenlin Yao, Yao Liu, Weizhi Zhang, Qin Lu, Liang Qiu, Changlong Yu, Puyang Xu, Chao Zhang, Bing Yin, Hyokun Yun, and Lihong Li. WebAgent-R1: Training web agents via end-to-end multi-turn reinforcement learning. In *EMNLP*, pp. 7909–7928, 2025.

Qiucheng Wu, Handong Zhao, Michael Saxon, Trung Bui, William Yang Wang, Yang Zhang, and Shiyu Chang. VSP: Diagnosing the dual challenges of perception and reasoning in spatial planning tasks for MLLMs. In *ICCV*, pp. 2270–2280, 2025.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In *NeurIPS*, volume 37, pp. 52040–52094, 2024.

Yutao Xie, Nathaniel Thomas, Nicklas Hansen, Yang Fu, Li Erran Li, and Xiaolong Wang. TIPS: Turn-level information-potential reward shaping for search-augmented LLMs. In *ICLR*, pp. 156549–156584, 2026.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. In *NeurIPS*, volume 35, pp. 20744–20757, 2022.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. In *NeurIPS*, volume 38, 2025.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *ICLR*, pp. 15585–15606, 2024a.

Yifei Zhou, Andrea Zanette, Jiayi Pan, Sergey Levine, and Aviral Kumar. ArCHer: Training language model agents via hierarchical multi-turn RL. In *ICML*, volume 235, pp. 62178–62209, 2024b.

Algorithm 1 Training Agentic Models with GHGPO

Require: Initial policy π_θ, reference policy π_{ref}, task distribution $p(\mathcal{X})$, distance discount factor ω, advantage weights $\lambda_E, \lambda_d, \lambda_h$, clipping parameter ϵ, KL penalty β, rollout group size M, maximum step, i.e. maximum interaction budget T

for each training iteration **do**
  Update the rollout policy: $\theta_{\text{old}} \leftarrow \theta$
  // Multi-step rollout phase
  Sample a task $x \sim p(\mathcal{X})$ and initialize M identical environments
  for $t = 1$ to T **do**
    Sample actions $\{a_t^m \sim \pi_{\theta_{\text{old}}}(\cdot \mid s_t^m, x)\}_{m=1}^M$
    Execute the actions and observe the next states $\{s_{t+1}^m\}_{m=1}^M$
  end for
  // Graph construction and transition estimation
  Construct the aggregated state-transition graph $\mathcal{G} = (\mathcal{S}, \mathcal{E})$ by Section 3
  Construct the policy-induced transition matrix $\widehat{\mathbf{P}}_\pi$ by Section 4.2
  Partition the states into $\mathcal{R}$, $\mathcal{S}^-$, *and* $\mathcal{S}^+$
  // Distance-based graph estimation
  Compute the distance $d(s)$ for each state $s \in \mathcal{S}$ using Eq. (2)
  // Finite-budget graph-hit estimation
  Compute the finite-budget graph-hit probability $h_\pi^B(s_i)$ *for each state* $s_i \in \mathcal{R}$ *using Eq. (11)*
  Set $h_\pi^B(s) = 1$ *for* $s \in \mathcal{S}^+$ *and* $h_\pi^B(s) = 0$ *for* $s \in \mathcal{S}^-$
  // State grouping phase
  Build state-level transition groups $G(s)$ according to Eq. (3)
  // Advantage computation phase
  Compute the trajectory-level outcome advantages $A_E(\tau_m)$ for $m = 1, \ldots, M$ using Eq. (1)
  Compute the distance-based process advantages $A_d(s_t^m, a_t^m, s_{t+1}^m)$ using Eq. (4)
  Compute the finite-budget hit-based process advantages $A_h^B(s_t^m, a_t^m, s_{t+1}^m)$ *using Eq. (14)*
  Combine the three advantage components using Eq. (15)
  // Policy update phase
  Update policy π_θ by maximizing Eq. (16)
end for

APPENDIX

A PSEUDO CODE

Algorithm 1 summarizes the complete training procedure of GHGPO. The training pipeline is organized into several stages, whose titles are shown in gray, while the additional operations introduced by GHGPO are highlighted in italics. GHGPO first collects grouped multi-step rollouts, aggregates them into a state-transition graph, estimates both structural distances and policy-induced finite-budget graph-hit probabilities, and then combines the outcome-, distance-, and hit-based advantages for policy optimization. Since these computations reuse the original rollout trajectories and involve only empirical transition estimation, finite-step recursion, and state-group normalization, GHGPO requires neither an auxiliary critic model nor additional environment interactions.

B PROOF OF PROPOSITION 1

Proposition 1 (Finite-budget graph-hit probability)**.** *Let the initial state be* s_1 *and the maximum number of interaction steps be* T*. For any intermediate state* $s_i \in \mathcal{R}$*, its normalized visiting step distribution is*

$$q_\pi(k \mid s_i) = \frac{\mu_k(s_i)}{\sum_{\ell=0}^{T} \mu_\ell(s_i)}, \qquad \mu_k(s_i) = \left(\left[\widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{R}} \right]^k \right)_{s_1, s_i}, \qquad k = 0, \ldots, T, \tag{10}$$

where $\mu_k(s_i)$ *denotes the probability of visiting state* s_i *at step* k *in the empirical transition process defined by* $\widehat{\mathbf{P}}_\pi$*. Then, applying the law of total probability over the partition induced by the visiting*

step, the finite-budget graph-hit probability of state s_i ($h^B_\pi(s_i)$) is calculated by marginalizing over its visiting step distribution $K \sim q_\pi(\cdot \mid s_i)$:

$$h^B_\pi(s_i) = \mathbb{E}_{K\sim q_\pi(\cdot|s_i)}\left[h^{T-K}_{\pi,\mathcal{R}}(s_i)\right] = \frac{\sum_{k=0}^{T}\mu_k(s_i)\,h^{T-k}_{\pi,\mathcal{R}}(s_i)}{\sum_{k=0}^{T}\mu_k(s_i)}. \tag{11}$$

Proof. Let the initial state be s_1, and let the maximum number of interaction steps be T. For any intermediate state $s_i \in \mathcal{R}$, let the probability of visiting state s_i after k transitions from s_1 be

$$\mu_k(s_i) = \left(\left[\widehat{\mathbf{P}}_{\pi,\mathcal{R}\mathcal{R}}\right]^k\right)_{s_1,s_i}, \qquad k = 0,\dots,T, \tag{17}$$

Since s_i is a state actually observed in the current grouped rollouts, there exists at least one finite path from s_1 to s_i with positive transition probability. Therefore, there exists some $k \in \{0,\dots,T\}$ such that $\mu_k(s_i) > 0$, and hence

$$\sum_{\ell=0}^{T}\mu_\ell(s_i) > 0. \tag{18}$$

Therefore, by normalizing the visitation probabilities over the finite horizon, we obtain the visiting step distribution of state s_i:

$$q_\pi(k \mid s_i) = \frac{\mu_k(s_i)}{\sum_{\ell=0}^{T}\mu_\ell(s_i)}, \qquad k = 0,\dots,T. \tag{19}$$

Here, $q_\pi(k \mid s_i)$ denotes the probability that, when a visitation instance of state s_i is sampled under the finite-horizon visitation measure, the visitation occurs at step k. By definition,

$$\sum_{k=0}^{T} q_\pi(k \mid s_i) = 1. \tag{20}$$

When state s_i is visited at step k, since the maximum number of interaction steps is T, the remaining interaction budget is $T-k$. By the definition of $h^m_{\pi,\mathcal{R}}(s_i)$, the success probability of reaching the success state set $\mathcal{S}^+$ from state s_i within this remaining budget is

$$h^{T-k}_{\pi,\mathcal{R}}(s_i). \tag{21}$$

Therefore, by marginalizing over the visiting step of state s_i, its finite-budget graph-hit probability can be written as

$$h^B_\pi(s_i) = \mathbb{E}_{K\sim q_\pi(\cdot|s_i)}\left[h^{T-K}_{\pi,\mathcal{R}}(s_i)\right]. \tag{22}$$

Expanding this expectation gives

$$h^B_\pi(s_i) = \sum_{k=0}^{T} q_\pi(k \mid s_i) h^{T-k}_{\pi,\mathcal{R}}(s_i) \tag{23}$$

$$= \sum_{k=0}^{T} \frac{\mu_k(s_i)}{\sum_{\ell=0}^{T}\mu_\ell(s_i)} h^{T-k}_{\pi,\mathcal{R}}(s_i) \tag{24}$$

$$= \frac{\sum_{k=0}^{T}\mu_k(s_i)h^{T-k}_{\pi,\mathcal{R}}(s_i)}{\sum_{\ell=0}^{T}\mu_\ell(s_i)}. \tag{25}$$

This proves the proposition. □

C Experimental Details

C.1 Benchmark and Evaluation Details

ALFWorld. ALFWorld (Shridhar et al., 2021) is a text-based embodied household environment for long-horizon decision making. It contains 3,827 task instances spanning six task categories: Pick & Place, Clean & Place, Cool & Place, Examine in Light, Heat & Place, and Pick Two & Place. In each episode, the agent receives a natural-language goal and interacts with the environment through textual observations and admissible actions. An episode is considered successful when the specified household task is completed. We report the success rate for each task category and the overall success rate across all categories.

WebShop. WebShop (Yao et al., 2022) is a text-based interactive shopping environment containing approximately 1.18 million products and 12,087 shopping instructions. Given a user instruction specifying desired product attributes, the agent interacts with the simulated shopping website to search, navigate, and purchase a suitable product. We report both the average task score, which measures how well the selected product satisfies the requested attributes,and the task success rate, where success requires the purchased product to satisfy all attributes specified in the user instruction.

Sokoban. Sokoban (Schrader, 2018) is a vision-based interactive puzzle environment requiring spatial reasoning and multi-step planning. We use the 6×6 setting, where the agent observes the current game state as an image and sequentially moves the player to push all boxes onto their target locations. An episode is successful when all boxes are placed on the targets. Because Sokoban instances are randomly generated rather than drawn from a fixed validation split, we evaluate performance by the average success rate over generated episodes.

C.2 Baseline Details

GPT-4o. GPT-4o (OpenAI, 2024) is a closed-source large language model included as a strong reference baseline for multi-turn agentic tasks.

Gemini-2.5-Pro. Gemini-2.5-Pro (Gemini Team, Google, 2023) is another closed-source large language model used as a competitive reference baseline.

Base-model Prompting. We directly prompt the corresponding pretrained base model to interact with the environment without parameter updates or additional adaptation.

ReAct. ReAct (Yao et al., 2023) is a prompting-based agent framework that interleaves reasoning and acting during multi-turn environment interaction.

Reflexion. Reflexion (Shinn et al., 2023) improves agent decisions through self-reflection over previous interaction outcomes without parameter updates.

PPO. PPO (Schulman et al., 2017) is a standard actor–critic policy optimization method that relies on a learned value function for advantage estimation.

RLOO. RLOO (Kool et al., 2019; Ahmadian et al., 2024) is a critic-free policy optimization method that estimates trajectory-level advantages using a leave-one-out baseline over grouped rollouts.

GRPO. GRPO (Shao et al., 2024) is a critic-free group-based policy optimization method that estimates trajectory-level relative advantages from multiple sampled responses.

GiGPO. GiGPO (Feng et al., 2025b) extends group-based policy optimization with fine-grained process credit by forming step-level groups from shared intermediate states and estimating group-wise step-level advantages.

GraphGPO. GraphGPO (Cheng et al., 2026) aggregates grouped rollouts into a state-transition graph and assigns fine-grained process credit according to structural distance to success states.

C.3 Training and Implementation Details

Shared Training Settings. All experiments are implemented based on the `verl-agent` framework (Feng et al., 2025b). Across all benchmarks, the maximum response length is 512 tokens, the actor learning rate is 1×10^{-6}, the rollout temperature is 1.0, and the evaluation temperature is 0.4. The KL-divergence loss coefficient is set to 0.01. We use a rule-based reward of 10 for successful task completion and 0 otherwise, with a penalty of -0.1 for invalid actions. All group-based RL methods use a rollout group size of 8. For GHGPO, we set $\lambda_E = \lambda_d = \lambda_h = 1$ across all benchmarks and model scales.

For the method-specific configurations, PPO uses a critic with a learning rate of 1×10^{-5} and 128 independent rollout environments. GiGPO uses a discount factor of 0.95. GraphGPO uses a benchmark-specific distance discount factor ω, which is also used by the distance-based advantage component of GHGPO.

ALFWorld. We use Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct as the base models. The maximum prompt length is 2048 tokens, and each episode allows up to 50 environment steps. For group-based methods, each iteration samples 16 groups with 8 rollouts per group, resulting in $16 \times 8 = 128$ rollout environments. The mini-batch size is 256. Following GraphGPO, the environment observations are augmented with the agent's current location and currently held objects. The distance discount factor ω is set to 0.10 for GraphGPO and GHGPO.

WebShop. We use Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct as the base models. The maximum prompt length is 5120 tokens, and each episode is limited to 15 environment steps. For group-based methods, each iteration samples 16 groups with 8 rollouts per group, resulting in 128 rollout environments. The mini-batch size is 64. We use the official text-rich version of WebShop, which provides additional information for interactive elements such as buttons. The distance discount factor ω is set to 0.20 for GraphGPO and GHGPO.

Sokoban. We use Qwen2.5-VL-3B-Instruct as the base vision-language model. The maximum prompt length is 1024 tokens, and each episode is limited to 15 environment steps. Group-based methods use a group size of 8 with $16 \times 8 = 128$ rollout environments in total, and the mini-batch size is 64. The distance discount factor ω is set to 0.80 for GraphGPO and GHGPO.

Prompt Templates The benchmark-specific prompt templates for ALFWorld, WebShop, and Sokoban are shown in Figures 5, 6, and 7, respectively. Placeholders enclosed in curly braces are dynamically instantiated at each interaction step, and the visible interaction history is limited to the two most recent steps. The model generates reasoning within `<think>...</think>` tags and the selected action within `<action>...</action>` tags. For Sokoban, the current visual observation is provided through the `<image>` placeholder.

Computing Details. All experiments are trained for 150 iterations. Experiments with Qwen2.5-1.5B-Instruct and Qwen2.5-VL-3B-Instruct are conducted on 2 NVIDIA H100 GPUs with 80 GB memory each, while Qwen2.5-7B-Instruct experiments use 4 NVIDIA H100 GPUs with 80 GB memory each.

D Additional experimental results

D.1 Effect of Visitation-Time Marginalization

To examine the effect of visitation-time marginalization, we compare GHGPO with a variant that directly uses the trajectory-specific remaining budget for each rollout transition. Specifically, for each $(s_t^m, a_t^m, s_{t+1}^m)$, $1 \le t \le T$ and $1 \le m \le M$, the variant uses $h_{\pi,\mathcal{R}}^{T-t}(s_{t+1}^m)$ based on the actual position of the transition in its rollout, instead of the visitation-time-marginalized state-level estimate $h_\pi^B(s_{t+1}^m)$. All other components of GHGPO remain unchanged. As shown in Table 4, GHGPO

Prompt Template for ALFWorld

You are an expert agent operating in the ALFRED embodied Environment. Your task is to: {task description}. Prior to this step, you have already taken {step count} step(s). Below are the most recent {history length} observations and the corresponding actions you took: {action history}. You are now at step {current step} and your current observation is: {current observation}. Your admissible actions of the current situation are: [{admissible actions}]. Location: {current location}. Items in hand (status): {current holding}. History moving list by you: {history moving}. Now it's your turn to take an action. You should first reason step-by-step about the current situation. This reasoning process MUST be enclosed within <think> </think> tags. Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Figure 5: Prompt template used for ALFWorld experiments.

Prompt Template for WebShop

You are an expert autonomous agent operating in the WebShop e-commerce environment. Your task is to: {task description}. Prior to this step, you have already taken {step count} step(s). Below are the most recent {history length} observations and the corresponding actions you took: {action history}. You are now at step {current step} and your current observation is: {current observation}. Your admissible actions for the current situation are: [{admissible actions}].
Now it's your turn to take one action for the current step. You should first reason step-by-step about the current situation, then think carefully which admissible action best advances the shopping goal. This reasoning process MUST be enclosed within <think> </think> tags. Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Figure 6: Prompt template used for WebShop experiments.

achieves higher overall performance on both benchmarks, improving the ALFWorld success rate from 95.89% to 96.61% and the WebShop success rate from 80.98% to 84.63%. This may be because deriving the visiting step distribution from the aggregated empirical transition graph reduces reliance on sparse occurrence-level statistics and allows visitation-time weighting and downstream hitting probabilities to reuse transition evidence across grouped rollouts. These results therefore support constructing state-level finite-budget graph-hit credit from the aggregated visiting step distribution, rather than directly relying on the position of each individual visitation in its original trajectory and the corresponding remaining interaction budget.

D.2 Effect of Ignoring the Remaining Interaction Budget

Limitations of Infinite-Budget Credit. Although both $\mathbf{h}_\pi^\infty$ and $\mathbf{h}_\pi^B$ are based on the empirical state-transition graph induced by the current policy, the former ignores the remaining interaction budget and may therefore fail to preserve differences in finite-budget success probability across actions. We analyze this effect on WebShop with Qwen2.5-1.5B-Instruct. The left panel of Figure 8 compares the ratio of the within-state-group standard deviations of $\mathbf{h}_\pi^\infty$ and $\mathbf{h}_\pi^B$. This ratio drops rapidly after the early stage of training and remains substantially below 1 for most of training, indicating that infinite-budget graph-hit probabilities provide weaker discrimination among candidate actions from the same state. The right panel of Figure 8 further directly compares the process advantages induced by the two success probability estimates. For transitions with $A_h^B > 0$ and $A_h^B < 0$, respectively, we measure the proportions for which A_h^∞ is positive, zero, or negative. On average, 82.08% of positive finite-budget process advantages and 79.89% of negative finite-budget process advantages become zero under the infinite-budget estimate. This shows that ignoring the remaining interaction budget not only reduces within-group contrast in state success probabilities, but also causes many actions with clearly positive or negative credit under the finite-budget estimate

Prompt Template for Sokoban

You are an expert agent operating in the Sokoban environment. Your goal is to push all the boxes onto the target spots. Once all boxes are on the targets, you win!
Rules
You can only push boxes. You can't pull them, so plan ahead to avoid getting stuck.
You can't walk through or push boxes into walls.
To avoid traps, do not push boxes into corners or against walls where they can't be moved again.
Visual Elements in the Image:
Character: A small, green alien-like figure with two antennae and black eyes. It represents you.
Box: A yellow crate marked with an orange "X" across its front. It is the box you need to push.
Target: A black tile outlined in red, with a small red diamond shape in the center. It marks the destination where a box should be pushed.
Current Step
Your current observation is shown in the image: <image>
Your admissible actions are ["up", "down", "left", "right"].
Now it's your turn to make a move (choose ONE action only for the current step).
You should first reason step-by-step about the current situation — observe the positions of boxes and targets, plan a path to push a box toward a target, and avoid traps like corners or walls. This reasoning process MUST be enclosed within <think> </think> tags. Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Figure 7: Prompt template used for Sokoban experiments.

Table 4: Effect of visitation-time marginalization on finite-budget estimation. Both ALFWorld and WebShop use Qwen2.5-1.5B-Instruct. GHGPO marginalizes over visiting steps when estimating finite-budget graph-hit probabilities, while the variant directly uses the remaining interaction budget of each step within its corresponding rollout trajectory.

Method	ALFWorld							WebShop	
	Pick	Clean	Cool	Look	Heat	Pick2	All	Score	Succ.
GHGPO	93.61$_{(0.12)}$	**100.0**$_{(0.00)}$	**95.83**$_{(7.22)}$	**97.22**$_{(4.82)}$	**98.15**$_{(3.21)}$	**97.10**$_{(5.02)}$	**96.61**$_{(1.96)}$	**92.27**$_{(2.46)}$	**84.63**$_{(2.74)}$
w/o time marginalization	**96.39**$_{(2.28)}$	97.83$_{(3.08)}$	95.00$_{(5.66)}$	95.83$_{(5.90)}$	97.22$_{(3.93)}$	90.89$_{(2.00)}$	95.89$_{(1.10)}$	90.46$_{(0.56)}$	80.98$_{(2.96)}$

to lose their discriminative signal. Therefore, GHGPO uses the process advantage A_h^B induced by the finite-budget graph-hit probability for optimization.

Ablation of Infinite-Budget Graph-Hit Advantage. We further replace the finite-budget advantage A_h^B with its infinite-budget counterpart A_h^∞. As shown in Table 5, this replacement consistently reduces aggregate performance across ALFWorld, WebShop, and Sokoban, showing that ignoring the remaining interaction budget weakens the effectiveness of graph-hit credit. Nevertheless, the infinite-budget variant still outperforms GraphGPO on all aggregate metrics, suggesting that policy-

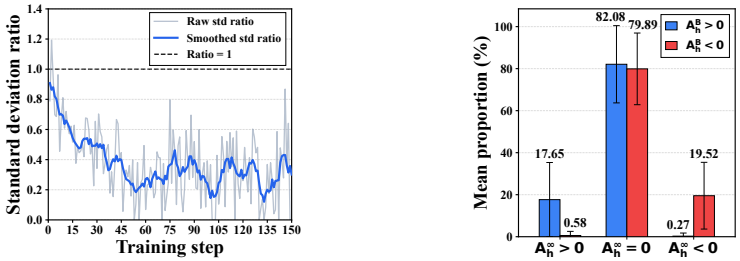


Figure 8: Effect of ignoring the remaining interaction budget in graph-hit credit assignment. **Left:** Infinite-budget graph-hit probability exhibits substantially lower within-group contrast than finite-budget graph-hit probability. **Right:** Most nonzero A_h^B values become zero under A_h^∞.

Table 5: Ablation study on ALFWorld, WebShop, and Sokoban. ALFWorld and WebShop use Qwen2.5-1.5B-Instruct, while Sokoban uses Qwen2.5-VL-3B-Instruct. Removing A_h^B reduces GHGPO to GraphGPO under the same experimental configuration, while $A_h^B \to A_h^\infty$ replaces the finite-budget graph-hit advantage with its infinite-budget counterpart.

Method	ALFWorld							WebShop		Sokoban
	Pick	Clean	Cool	Look	Heat	Pick2	All	Score	Succ.	Succ.
GHGPO	$93.61_{(0.12)}$	$\mathbf{100.0}_{(0.00)}$	$\mathbf{95.83}_{(7.22)}$	$\mathbf{97.22}_{(4.82)}$	$\mathbf{98.15}_{(3.21)}$	$\mathbf{97.10}_{(5.02)}$	$\mathbf{96.61}_{(1.96)}$	$\mathbf{92.27}_{(2.46)}$	$\mathbf{84.63}_{(2.74)}$	$\mathbf{88.28}_{(1.35)}$
$-A_h^B$	$95.15_{(1.62)}$	$\mathbf{100.0}_{(0.00)}$	$85.26_{(2.58)}$	$85.71_{(5.83)}$	$96.30_{(2.61)}$	$93.65_{(2.24)}$	$92.71_{(1.32)}$	$89.29_{(1.48)}$	$78.65_{(3.86)}$	$86.98_{(0.73)}$
$A_h^B \to A_h^\infty$	$\mathbf{96.77}_{(3.23)}$	$98.67_{(2.31)}$	$91.38_{(0.80)}$	$\mathbf{97.22}_{(4.82)}$	$98.04_{(3.40)}$	$89.24_{(9.82)}$	$94.79_{(1.19)}$	$90.16_{(1.45)}$	$82.81_{(2.71)}$	$87.05_{(1.41)}$

Table 6: Ablation study on ALFWorld with Qwen2.5-7B-Instruct. The variant without A_h^B reduces to GraphGPO under the same experimental configuration.

Method	ALFWorld						
	Pick	Clean	Cool	Look	Heat	Pick2	All
GHGPO	$\mathbf{100.00}_{(0.00)}$	$\mathbf{100.00}_{(0.00)}$	$\mathbf{100.00}_{(0.00)}$	$\mathbf{96.97}_{(5.25)}$	$\mathbf{100.00}_{(0.00)}$	$93.04_{(3.60)}$	$\mathbf{98.69}_{(0.90)}$
$-A_d$	$99.10_{(1.56)}$	$97.15_{(2.48)}$	$95.71_{(4.35)}$	$92.31_{(13.33)}$	$93.69_{(5.93)}$	$\mathbf{96.33}_{(3.27)}$	$96.35_{(3.16)}$
$-A_h^B$	$\mathbf{100.0}_{(0.00)}$	$\mathbf{100.0}_{(0.00)}$	$91.40_{(1.50)}$	$92.86_{(5.83)}$	$94.44_{(0.00)}$	$92.06_{(2.24)}$	$95.31_{(1.10)}$

dependent graph-hit probabilities are informative by themselves, while explicitly accounting for the remaining interaction budget provides a stronger process-credit signal.

D.3 Additional Ablation on Qwen2.5-7B-Instruct

To examine whether the contributions of the two process-level advantages persist at a larger model scale, we repeat the component ablation on ALFWorld with Qwen2.5-7B-Instruct. As shown in Table 6, removing either A_d or A_h^B reduces the overall success rate from 98.69% to 96.35% and 95.31%, respectively. The larger degradation caused by removing A_h^B is consistent with the 1.5B results, while the full GHGPO again achieves the best overall performance. These results show that the complementary roles of structural efficiency and finite-budget success probability persist as the base model scales up.

D.4 Evolution of the Empirical Transition Graph

We examine how the empirical transition structure used by GHGPO evolves during training and whether grouped rollouts provide sufficient repeated transition evidence for estimating the empirical transition graph. Figure 9 reports the state-group size, state out-degree, and edge sample count at different stages of training. As training progresses, state groups become increasingly populated: their mean size grows from 4.1 at step 10 to 7.0 at step 70 and 8.3 at step 150, while the proportion of singleton groups decreases from 43.8% to 20.1%. Meanwhile, the mean out-degree decreases from 2.0 to 1.5, suggesting that rollout behavior becomes less dispersed across alternative successor states and increasingly concentrates on a smaller set of choices as the policy is optimized. The empirical support for individual transitions also increases over training: the mean number of samples per distinct edge rises from 2.0 to 5.1, while the proportion of edges observed only once drops from 66.3% to 33.2%. Together with the increasing state-group size, these repeated observations provide stronger empirical support for estimating the transition probabilities in $\widehat{\mathbf{P}}_\pi$, potentially reducing finite-sample noise and yielding more stable finite-budget graph-hit estimates.

D.5 Effect of Rollout Group Size

Table 7 examines the effect of rollout group size on WebShop. GHGPO consistently achieves the best performance across all evaluated group sizes. Even with a small group size ($M = 4$), GHGPO slightly outperforms GraphGPO, indicating that finite-budget graph-hit credit remains effective with relatively limited transition evidence. As the group size increases, the improvement becomes more pronounced: GHGPO outperforms GraphGPO by 5.88 percentage points at $M = 6$ and 5.98 points at $M = 8$. This trend is consistent with the design of GHGPO: larger rollout groups provide more

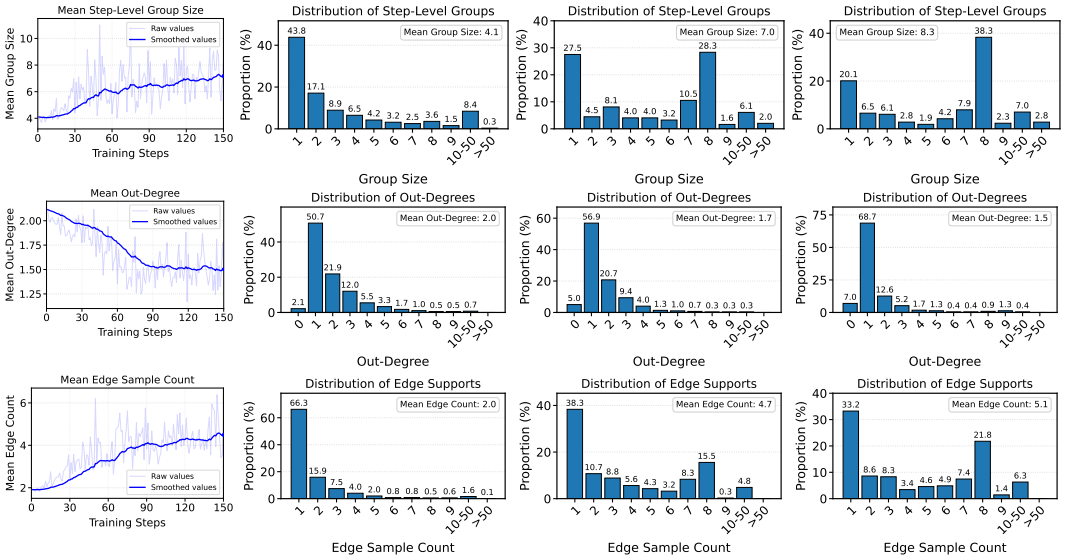


Figure 9: Evolution of the empirical transition structure during training with Qwen2.5-1.5B-Instruct on ALFWorld. Rows show state-group size, state out-degree, and edge sample count, respectively. The first column shows their training dynamics, while the remaining columns show distributions at steps 10, 70, and 150. State groups become increasingly populated and transition edges receive more repeated observations as training progresses, while the graph becomes less branched.

Table 7: The test performance of GRPO, GiGPO, GraphGPO and GHGPO under different group sizes on WebShop. We report the average success rate (%). All results are averaged over 3 random seeds. The best performance in each row is highlighted in bold.

Group size	GRPO	GiGPO	GraphGPO	GHGPO
$M = 4$	$65.23_{(1.99)}$	$69.90_{(2.19)}$	$74.06_{(2.88)}$	$\mathbf{74.74}_{(3.85)}$
$M = 6$	$66.41_{(2.78)}$	$73.34_{(1.98)}$	$74.58_{(3.00)}$	$\mathbf{80.46}_{(1.57)}$
$M = 8$	$71.35_{(2.05)}$	$73.83_{(2.30)}$	$78.65_{(3.86)}$	$\mathbf{84.63}_{(2.74)}$

transition observations for constructing the empirical transition graph, which can strengthen the estimation of policy-dependent finite-budget graph-hit probability.

E Case Study

ALFWorld. We present a successful interaction trajectory produced by an agent trained with GHGPO on ALFWorld. The prompt and model response at each interaction step are shown below, illustrating how the trained agent completes a multi-turn embodied task through sequential decisions.

Prompt (ALFWorld, Step 1)

You are an expert agent operating in the ALFRED Embodied Environment.
Your current observation is: -= Welcome to TextWorld, ALFRED! =-\n\nYou are in the middle of a room. Looking quickly around you, you see a bed 1, a desk 1, a drawer 6, a drawer 5, a drawer 4, a drawer 3, a drawer 2, a drawer 1, a garbagecan 1, a shelf 2, a shelf 1, and a sidetable 1. \n\nYour task is to: put two book in desk. Location: middle of a room. Items in hand (status): nothing.
Your admissible actions of the current situation are: ['go to bed 1', 'go to desk 1', 'go to drawer 1', 'go to drawer 2', 'go to drawer 3', 'go to drawer 4', 'go to drawer 5', 'go to drawer 6', 'go to garbagecan 1', 'go to shelf 1', 'go to shelf 2', 'go to sidetable 1', 'inventory', 'look'].

Now it's your turn to take an action. You should first reason step-by-step about the current situation. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (ALFWorld, Step 1)

<think>\nThe task is to put two books in the desk. The current location is the middle of the room, and the items in hand are nothing. There are various locations and items to consider. The first step is to find books.\n</think>\n\n<action>go to bed 1</action>

Prompt (ALFWorld, Step 2)

You are an expert agent operating in the ALFRED Embodied Environment. Your task is to: put two book in desk.
Prior to this step, you have already taken 1 step(s). Below are the most recent 1 observations and the corresponding actions you took: [Observation 1: '-= Welcome to TextWorld, ALFRED! =-\n\nYou are in the middle of a room. Looking quickly around you, you see a bed 1, a desk 1, a drawer 6, a drawer 5, a drawer 4, a drawer 3, a drawer 2, a drawer 1, a garbagecan 1, a shelf 2, a shelf 1, and a sidetable 1.\n\nYour task is to: put two book in desk. Location: middle of a room. Items in hand (status): nothing. ', Action 1: 'go to bed 1']
You are now at step 2 and your current observation is: You arrive at bed 1. On the bed 1, you see a book 2, a book 1, a cellphone 2, a laptop 1, a pillow 1, and a teddybear 1. Location: bed 1. Items in hand (status): nothing.
Your admissible actions of the current situation are: ['examine bed 1', 'go to desk 1', 'go to drawer 1', 'go to drawer 2', 'go to drawer 3', 'go to drawer 4', 'go to drawer 5', 'go to drawer 6', 'go to garbagecan 1', 'go to shelf 1', 'go to shelf 2', 'go to sidetable 1', 'inventory', 'look', 'take book 1 from bed 1', 'take book 2 from bed 1', 'take cellphone 2 from bed 1', 'take laptop 1 from bed 1', 'take pillow 1 from bed 1', 'take teddybear 1 from bed 1'].

Now it's your turn to take an action.
You should first reason step-by-step about the current situation. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (ALFWorld, Step 2)

<think>\nThe current situation is that I am at bed 1. I have two books (book 1 and book 2) and other items like a cellphone, laptop, pillow, and a teddybear. My task is to put two books in the desk. The books are on the bed, and the desk is the location I need to put them.\n</think>\n\n<action>take book 1 from bed 1</action>

Prompt (ALFWorld, Step 3)

You are an expert agent operating in the ALFRED Embodied Environment. Your task is to: put two book in desk.
Prior to this step, you have already taken 2 step(s). Below are the most recent 2 observations and the corresponding actions you took: [Observation 1: '-= Welcome to TextWorld, ALFRED! =-\n\nYou are in the middle of a room. Looking quickly around you, you see a bed 1, a desk 1, a drawer 6, a drawer 5, a drawer 4, a drawer 3, a drawer 2, a drawer 1, a garbagecan 1, a shelf 2, a shelf 1, and a sidetable 1.\n\nYour task is to: put two book in desk. Location: middle of a room. Items in hand (status): nothing. ', Action 1: 'go to bed 1']\n[Observation 2: 'You arrive at bed 1. On the bed 1, you see a book 2, a book 1, a cellphone 2, a laptop 1, a pillow 1, and a teddybear 1. Location: bed 1. Items in hand (status): nothing. ', Action 2: 'take book 1 from bed 1']
You are now at step 3 and your current observation is: You pick up the book 1 from the bed 1. Location: bed 1. Items in hand (status): book 1(unprocessed).
Your admissible actions of the current situation are: ['examine bed 1', 'examine book 1', 'go to desk 1', 'go to drawer 1', 'go to drawer 2', 'go to drawer 3', 'go to drawer 4', 'go to drawer 5', 'go to drawer 6', 'go to garbagecan 1', 'go to shelf 1', 'go to shelf 2', 'go to sidetable 1', 'inventory', 'look', 'move book 1 to bed 1'].

Now it's your turn to take an action.
You should first reason step-by-step about the current situation. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (ALFWorld, Step 3)

<think>\nThe current situation is that I have taken the book 1 from the bed 1. My next step is to put this book in the desk 1.\n</think>\n\n<action>go to desk 1</action>

Prompt (ALFWorld, Step 4)

You are an expert agent operating in the ALFRED Embodied Environment. Your task is to: put two book in desk.
Prior to this step, you have already taken 3 step(s). Below are the most recent 2 observations and the corresponding actions you took: [Observation 2: 'You arrive at bed 1. On the bed 1, you see a book 2, a book 1, a cellphone 2, a laptop 1, a pillow 1, and a teddybear 1. Location: bed 1. Items in hand (status): nothing. ', Action 2: 'take book 1 from bed 1']\n[Observation 3: 'You pick up the book 1 from the bed 1. Location: bed 1. Items in hand (status): book 1(unprocessed). ', Action 3: 'go to desk 1']
You are now at step 4 and your current observation is: You arrive at desk 1. On the desk 1, you see a keychain 2, a keychain 1, a mug 2, a pen 2, a tissuebox 3, a tissuebox 2, and a tissuebox 1. Location: desk 1. Items in hand (status): book 1(unprocessed).
Your admissible actions of the current situation are: ['examine book 1', 'examine desk 1', 'go to bed 1', 'go to drawer 1', 'go to drawer 2', 'go to drawer 3', 'go to drawer 4', 'go to drawer 5', 'go to drawer 6', 'go to garbagecan 1', 'go to shelf 1', 'go to shelf 2', 'go to sidetable 1', 'inventory', 'look', 'move book 1 to desk 1'].

Now it's your turn to take an action.
You should first reason step-by-step about the current situation. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (ALFWorld, Step 4)

<think>\nThe current situation is: I have a book 1 in hand and I am at desk 1. I need to put the book 1 in the desk 1.\n</think>\n\n<action>move book 1 to desk 1</action>

Prompt (ALFWorld, Step 5)

You are an expert agent operating in the ALFRED Embodied Environment. Your task is to: put two book in desk.
Prior to this step, you have already taken 4 step(s). Below are the most recent 2 observations and the corresponding actions you took: [Observation 3: 'You pick up the book 1 from the bed 1. Location: bed 1. Items in hand (status): book 1(unprocessed). ', Action 3: 'go to desk 1']\n[Observation 4: 'You arrive at desk 1. On the desk 1, you see a keychain 2, a keychain 1, a mug 2, a pen 2, a tissuebox 3, a tissuebox 2, and a tissuebox 1. Location: desk 1. Items in hand (status): book 1(unprocessed). ', Action 4: 'move book 1 to desk 1']
You are now at step 5 and your current observation is: You move the book 1 to the desk 1. Location: desk 1. Items in hand (status): nothing. History moving list by you: book 1(unprocessed) from bed 1 move to desk 1;
Your admissible actions of the current situation are: ['examine desk 1', 'go to bed 1', 'go to drawer 1', 'go to drawer 2', 'go to drawer 3', 'go to drawer 4', 'go to drawer 5', 'go to drawer 6', 'go to garbagecan 1', 'go to shelf 1', 'go to shelf 2', 'go to sidetable 1', 'inventory', 'look', 'take book 1 from desk 1', 'take keychain 1 from desk 1', 'take keychain 2 from desk 1', 'take mug 2 from desk 1', 'take pen 2 from desk 1', 'take tissuebox 1 from desk 1', 'take tissuebox 2 from desk 1', 'take tissuebox 3 from desk 1'].

Now it's your turn to take an action.
You should first reason step-by-step about the current situation. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (ALFWorld, Step 5)

<think>\nThe current situation is that I have just moved a book from the bed 1 to the desk 1. The book is now on the desk 1. I need to put another book in the desk 1.\n</think>\n\n<action>go to bed 1</action>

Prompt (ALFWorld, Step 6)

You are an expert agent operating in the ALFRED Embodied Environment. Your task is to: put two book in desk.
Prior to this step, you have already taken 5 step(s). Below are the most recent 2 observations and the corresponding actions you took: [Observation 4: 'You arrive at desk 1. On the desk 1, you see a keychain 2, a keychain 1, a mug 2, a pen 2, a tissuebox 3, a tissuebox 2, and a tissuebox 1. Location: desk 1. Items in hand (status): book 1(unprocessed). ', Action 4: 'move book 1 to desk 1']\n[Observation 5: 'You move the book 1 to the desk 1. Location: desk 1. Items in hand (status): nothing. History moving list by you: book 1(unprocessed) from bed 1 move to desk 1;', Action 5: 'go to bed 1']
You are now at step 6 and your current observation is: You arrive at bed 1. On the bed 1, you see a book 2, a cellphone 2, a laptop 1, a pillow 1, and a teddybear 1. Location: bed 1. Items in hand (status): nothing. History moving list by you: book 1(unprocessed) from bed 1 move to desk 1;
Your admissible actions of the current situation are: ['examine bed 1', 'go to desk 1', 'go to drawer 1', 'go to drawer 2', 'go to drawer 3', 'go to drawer 4', 'go to drawer 5', 'go to drawer 6', 'go to garbagecan 1', 'go to shelf 1', 'go to shelf 2', 'go to sidetable 1', 'inventory', 'look', 'take book 2 from bed 1', 'take cellphone 2 from bed 1', 'take laptop 1 from bed 1', 'take pillow 1 from bed 1', 'take teddybear 1 from bed 1'].

Now it's your turn to take an action.
You should first reason step-by-step about the current situation. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (ALFWorld, Step 6)

<think>\nThe current situation is that I am at bed 1, and I have already moved book 1 to desk 1. Now, I need to put another book in the desk. I have a book 2 on the bed 1, and I need to take it and put it in the desk 1.\n</think>\n\n<action>take book 2 from bed 1</action>

Prompt (ALFWorld, Step 7)

You are an expert agent operating in the ALFRED Embodied Environment. Your task is to: put two book in desk.
Prior to this step, you have already taken 6 step(s). Below are the most recent 2 observations and the corresponding actions you took: [Observation 5: 'You move the book 1 to the desk 1. Location: desk 1. Items in hand (status): nothing. History moving list by you: book 1(unprocessed) from bed 1 move to desk 1;', Action 5: 'go to bed 1']\n[Observation 6: 'You arrive at bed 1. On the bed 1, you see a book 2, a cellphone 2, a laptop 1, a pillow 1, and a teddybear 1. Location: bed 1. Items in hand (status): nothing. History moving list by you: book 1(unprocessed) from bed 1 move to desk 1;', Action 6: 'take book 2 from bed 1']
You are now at step 7 and your current observation is: You pick up the book 2 from the bed 1. Location: bed 1. Items in hand (status): book 2(unprocessed). History moving list by you: book 1(unprocessed) from bed 1 move to desk 1;
Your admissible actions of the current situation are: ['examine bed 1', 'examine book 2', 'go to desk 1', 'go to drawer 1', 'go to drawer 2', 'go to drawer 3', 'go to drawer 4', 'go to drawer 5', 'go to drawer 6', 'go to garbagecan 1', 'go to shelf 1', 'go to shelf 2', 'go to sidetable 1', 'inventory', 'look', 'move book 2 to bed 1'].

Now it's your turn to take an action.
You should first reason step-by-step about the current situation. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (ALFWorld, Step 7)

<think>\nThe current situation is that I have picked up book 2 from the bed 1. I need to put this book in the desk 1. The book 2 is currently in the bed 1, and I need to move it to the desk 1.\n</think>\n\n<action>go to desk 1</action>

WebShop. We further present a successful WebShop trajectory generated by an agent trained with GHGPO. The task requires satisfying multiple shopping constraints on color, shape, size, and price. The trajectory below shows the complete six-step interaction, from the initial product search to attribute selection and the final purchase action.

Prompt (WebShop, Step 1)

You are an expert autonomous agent operating in the WebShop e-commerce environment.
Your task is to: Find me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars.
Your current observation is: WebShop\nInstruction: \nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Search [button_]\n.
Your admissible actions of the current situation are: ['search[<your query>]', 'click[search]'].

Now it's your turn to take one action for the current step.
You should first reason step-by-step about the current situation, then think carefully which admissible action best advances the shopping goal. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (WebShop, Step 1)

<think>\nTo find the machine wash men's t-shirts and tanks with the specified features, I need to search for them.\n</think>\n<action>search[men's t-shirts & tanks with polyester heathers, royal blue, size: large, fit type: classic fit, color: royal blue, size: large, price: 40.00 USD]</action>

Prompt (WebShop, Step 2)

You are an expert autonomous agent operating in the WebShop e-commerce environment.
Your task is to: Find me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars.
Prior to this step, you have already taken 1 step(s). Below are the most recent 1 observations and the corresponding actions you took: [Observation 1: 'WebShop\nInstruction: \nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Search [button_]\n', Action 1: 'search[men's t-shirts & tanks with polyester heathers, royal blue, size: large, fit type: classic fit, color: royal blue, size: large, price: 40.00 usd]']
You are now at step 2 and your current observation is: Instruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\nPage 1 (Total results: 50)\n[button] Next > [button_]\n\n[button] B09RVF4JP1 [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\n$19.99\n\n[button] B07JVVDJ6L [button_]\nCanadian Tuxedo T Shirt - Novelty Denim Tux T-Shirt\n$19.99\n\n[button] B09P39QN2W [button_]\nI Tackled 100 Day Of School Football Boy 100th Day School T-Shirt\n$16.99\n\n[button] B0969G2DH8 [button_]\nThe cool twins just showed up T-Shirt\n$17.99\n\n[button] B07T3VMGDG [button_]\nTuxedo Cat 4th of July Hat Patriotic Gift Adults Kids Raglan Baseball Tee\n$23.99\n\n[button] B09KLQLLT2 [button_]\nLong Sleeve Superhero T Shirt Tank Top Mens Compression Shirt Men Workout Fitness Gym Shirt\n$19.99\n\n[button] B07HRFSNL4 [button_]\nNautica Men's Solid Crew Neck Short-Sleeve Pocket T-Shirt\n$16.05 to $40.98\n\n[button] B09GL561XH [button_]\nDisney Mickey And Friends Happy Birthday Mickey Confetti T-Shirt\n$22.99\n\n[button] B09NNMV9LN [button_]\nJSPOYOU Mens Short Sleeve Crewneck 3D Graphic Tunic Shirts Big & Tall Tie Dye Summer Top Basic Designed Classic Cotton Shirt\n$3.99 to $10.99\n\n[button] B09QQP3356 [button_]\nHAUKLIE Men's Sports Waffle Ribbed Polo Shirts Summer Short Sleeve Cotton Muscle Quarter-Zip Henley T-Shirt Tunics Tops\n$10.99\n.
Your admissible actions of the current situation are: ['click[back to search]', 'click[next >]', 'click[b09rvf4jp1]', 'click[b07jvvdj6l]', 'click[b09p39qn2w]', 'click[b0969g2dh8]', 'click[b07t3vmgdg]', 'click[b09klqllt2]', 'click[b07hrfsnl4]', 'click[b09gl561xh]', 'click[b09nnmv9ln]', 'click[b09qqp3356]'].

Now it's your turn to take one action for the current step.
You should first reason step-by-step about the current situation, then think carefully which admissible action best advances the shopping goal. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (WebShop, Step 2)

<think>\nTo find the machine wash men's t-shirts and tanks with the specified characteristics, I need to narrow down the search to only those items that meet all of the given criteria.\n</think>\n<action>click[b09rvf4jp1]</action>

Prompt (WebShop, Step 3)

You are an expert autonomous agent operating in the WebShop e-commerce environment.
Your task is to: Find me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars.
Prior to this step, you have already taken 2 step(s). Below are the most recent 2 observations and the corresponding actions you took: [Observation 1: 'WebShop\nInstruction: \nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Search [button_]\n', Action 1: 'search[men's t-shirts & tanks with polyester heathers, royal blue, size: large, fit type: classic fit, color: royal blue, size: large, price: 40.00 usd]']\n[Observation 2: 'Instruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\nPage 1 (Total results: 50)\n[button] Next > [button_]\n\n[button] B09RVF4JP1 [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\n$19.99\n\n[button] B07JVVDJ6L [button_]\nCanadian Tuxedo T Shirt - Novelty Denim Tux T-Shirt\n$19.99\n\n[button] B09P39QN2W [button_]\nI Tackled 100 Day Of School Football Boy 100th Day School T-Shirt\n$16.99\n\n[button] B0969G2DH8 [button_]\nThe cool twins just showed up T-Shirt\n$17.99\n\n[button] B07T3VMGDG [button_]\nTuxedo Cat 4th of July Hat Patriotic Gift Adults Kids Raglan Baseball Tee\n$23.99\n\n[button] B09KLQLLT2 [button_]\nLong Sleeve Superhero T Shirt Tank Top Mens Compression Shirt Men Workout Fitness Gym Shirt\n$19.99\n\n[button] B07HRFSNL4 [button_]\nNautica Men's Solid Crew Neck Short-Sleeve Pocket T-Shirt\n$16.05 to $40.98\n\n[button] B09GL561XH [button_]\nDisney Mickey And Friends Happy Birthday Mickey Confetti T-Shirt\n$22.99\n\n[button] B09NNMV9LN [button_]\nJSPOYOU Mens Short Sleeve Crewneck 3D Graphic Tunic Shirts Big & Tall Tie Dye Summer Top Basic Designed Classic Cotton Shirt\n$3.99 to $10.99\n\n[button] B09QQP3356 [button_]\nHAUKLIE Men's Sports Waffle Ribbed Polo Shirts Summer Short Sleeve Cotton Muscle Quarter-Zip Henley T-Shirt Tunics Tops\n$10.99\n', Action 2: 'click[b09rvf4jp1]']
You are now at step 3 and your current observation is: Instruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\n[button] < Prev [button_]\nfit type\n [button] men [button_]\n [button] women [button_]\ncolor\n [button] black [button_]\n [button] navy [button_]\n [button] dark heather [button_]\n [button] neon pink [button_]\n [button] royal blue [button_]\nsize\n [button] small [button_]\n [button] medium [button_]\n [button] large [button_]\n [button] x-large [button_]\n [button] xx-large [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\nPrice: $19.99\nRating: N.A.\n[button] Description [button_]\n[button] Features [button_]\n[button] Reviews [button_]\n[button] Buy Now [button_]\n.

Your admissible actions of the current situation are: ['click[back to search]', 'click[< prev]', 'click[description]', 'click[features]', 'click[reviews]', 'click[buy now]', 'click[men]', 'click[women]', 'click[black]', 'click[navy]', 'click[dark heather]', 'click[neon pink]', 'click[royal blue]', 'click[small]', 'click[medium]', 'click[large]', 'click[x-large]', 'click[xx-large]'].

Now it's your turn to take one action for the current step.
You should first reason step-by-step about the current situation, then think carefully which admissible action best advances the shopping goal. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (WebShop, Step 3)

<think>\nTo find the machine wash men's t-shirts and tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars, I need to refine the search to include only those specific criteria.\n</think>\n<action>click[men]</action>

Prompt (WebShop, Step 4)

You are an expert autonomous agent operating in the WebShop e-commerce environment.
Your task is to: Find me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars.
Prior to this step, you have already taken 3 step(s). Below are the most recent 2 observations and the corresponding actions you took: [Observation 2: 'Instruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\nPage 1 (Total results: 50)\n[button] Next > [button_]\n\n[button] B09RVF4JP1 [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\n$19.99\n\n[button] B07JVVDJ6L [button_]\nCanadian Tuxedo T Shirt - Novelty Denim Tux T-Shirt\n$19.99\n\n[button] B09P39QN2W [button_]\nI Tackled 100 Day Of School Football Boy 100th Day School T-Shirt\n$16.99\n\n[button] B0969G2DH8 [button_]\nThe cool twins just showed up T-Shirt\n$17.99\n\n[button] B07T3VMGDG [button_]\nTuxedo Cat 4th of July Hat Patriotic Gift Adults Kids Raglan Baseball Tee\n$23.99\n\n[button] B09KLQLLT2 [button_]\nLong Sleeve Superhero T Shirt Tank Top Mens Compression Shirt Men Workout Fitness Gym Shirt\n$19.99\n\n[button] B07HRFSNL4 [button_]\nNautica Men's Solid Crew Neck Short-Sleeve Pocket T-Shirt\n$16.05 to $40.98\n\n[button] B09GL561XH [button_]\nDisney Mickey And Friends Happy Birthday Mickey Confetti T-Shirt\n$22.99\n\n[button] B09NNMV9LN [button_]\nJSPOYOU Mens Short Sleeve Crewneck 3D Graphic Tunic Shirts Big & Tall Tie Dye Summer Top Basic Designed Classic Cotton Shirt\n$3.99 to $10.99\n\n[button] B09QQP3356 [button_]\nHAUKLIE Men's Sports Waffle Ribbed Polo Shirts Summer Short Sleeve Cotton Muscle Quarter-Zip Henley T-Shirt Tunics Tops\n$10.99\n', Action 2: 'click[b09rvf4jp1]']\n[Observation 3: 'Instruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\n[button] < Prev [button_]\nfit type\n [button] men [button_]\n [button] women [button_]\ncolor\n [button] black [button_]\n [button] navy [button_]\n [button] dark heather [button_]\n [button] neon pink [button_]\n [button] royal blue [button_]\nsize\n [button] small [button_]\n [button] medium [button_]\n [button] large [button_]\n [button] x-large [button_]\n [button] xx-large [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\nPrice: $19.99\nRating: N.A.\n[button] Description [button_]\n[button] Features [button_]\n[button] Reviews [button_]\n[button] Buy Now [button_]\n', Action 3: 'click[men]']
You are now at step 4 and your current observation is: You have clicked men.\nInstruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\n[button] < Prev [button_]\nfit type\n [clicked button] men [clicked button_]\n [button] women [button_]\ncolor\n [button] black [button_]\n [button] navy [button_]\n [button] dark heather [button_]\n [button] neon pink [button_]\n [button] royal blue [button_]\nsize\n [button] small [button_]\n [button] medium [button_]\n [button] large [button_]\n [button] x-large [button_]\n [button] xx-large [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\nPrice: $19.99\nRating: N.A.\n[button] Description [button_]\n[button] Features [button_]\n[button] Reviews [button_]\n[button] Buy Now [button_]\n.
Your admissible actions of the current situation are: ['click[back to search]', 'click[< prev]', 'click[description]', 'click[features]', 'click[reviews]', 'click[buy now]', 'click[men]', 'click[women]', 'click[black]', 'click[navy]', 'click[dark heather]', 'click[neon pink]', 'click[royal blue]', 'click[small]', 'click[medium]', 'click[large]', 'click[x-large]', 'click[xx-large]'].

Now it's your turn to take one action for the current step.
You should first reason step-by-step about the current situation, then think carefully which admissible action best advances the shopping goal. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (WebShop, Step 4)

<think>\nTo find the machine wash men's t-shirts and tanks with the specified features, I need to filter the results based on the color, fit type, size, and price criteria.\n</think>\n<action>click[royal blue]</action>

Prompt (WebShop, Step 5)

You are an expert autonomous agent operating in the WebShop e-commerce environment.
Your task is to: Find me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars.
Prior to this step, you have already taken 4 step(s). Below are the most recent 2 observations and the corresponding actions you took: [Observation 3: 'Instruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\n[button] < Prev [button_]\nfit type\n [button] men [button_]\n [button] women [button_]\ncolor\n [button] black [button_]\n [button] navy [button_]\n [button] dark heather [button_]\n [button] neon pink [button_]\n [button] royal blue [button_]\nsize\n [button] small [button_]\n [button] medium [button_]\n [button] large [button_]\n [button] x-large [button_]\n [button] xx-large [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\nPrice: $19.99\nRating: N.A.\n[button] Description [button_]\n[button] Features [button_]\n[button] Reviews [button_]\n[button] Buy Now [button_]\n', Action 3: 'click[men]']\n[Observation 4: 'You have clicked men.\nInstruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\n[button] < Prev [button_]\nfit type\n [clicked button] men [clicked button_]\n [button] women [button_]\ncolor\n [button] black [button_]\n [button] navy [button_]\n [button] dark heather [button_]\n [button] neon pink [button_]\n [button] royal blue [button_]\nsize\n [button] small [button_]\n [button] medium [button_]\n [button] large [button_]\n [button] x-large [button_]\n [button] xx-large [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\nPrice: $19.99\nRating: N.A.\n[button] Description [button_]\n[button] Features [button_]\n[button] Reviews [button_]\n[button] Buy Now [button_]\n', Action 4: 'click[royal blue]']
You are now at step 5 and your current observation is: You have clicked royal blue.\nYou have clicked men.\nInstruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\n[button] < Prev [button_]\nfit type\n [clicked button] men [clicked button_]\n [button] women [button_]\ncolor\n [button] black [button_]\n [button] navy [button_]\n [button] dark heather [button_]\n [button] neon pink [button_]\n [clicked button] royal blue [clicked button_]\nsize\n [button] small [button_]\n [button] medium [button_]\n [button] large [button_]\n [button] x-large [button_]\n [button] xx-large [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\nPrice: $19.99\nRating: N.A.\n[button] Description [button_]\n[button] Features [button_]\n[button] Reviews [button_]\n[button] Buy Now [button_]\n.

Your admissible actions of the current situation are: ['click[back to search]', 'click[< prev]', 'click[description]', 'click[features]', 'click[reviews]', 'click[buy now]', 'click[men]', 'click[women]', 'click[black]', 'click[navy]', 'click[dark heather]', 'click[neon pink]', 'click[royal blue]', 'click[small]', 'click[medium]', 'click[large]', 'click[x-large]', 'click[xx-large]'].

Now it's your turn to take one action for the current step.
You should first reason step-by-step about the current situation, then think carefully which admissible action best advances the shopping goal. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (WebShop, Step 5)

<think>\nTo find the desired items, I need to filter the options based on the given criteria:\n- Color: Royal Blue\n- Fit type: Classic Fit\n- Size: Large\n- Price lower than 40.00 dollars\n</think>\n<action>click[large]</action>

Prompt (WebShop, Step 6)

You are an expert autonomous agent operating in the WebShop e-commerce environment.
Your task is to: Find me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars.
Prior to this step, you have already taken 5 step(s). Below are the most recent 2 observations and the corresponding actions you took: [Observation 4: 'You have clicked men.\nInstruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\n[button] < Prev [button_]\nfit type\n [clicked button] men [clicked button_]\n [button] women [button_]\ncolor\n [button] black [button_]\n [button] navy [button_]\n [button] dark heather [button_]\n [button] neon pink [button_]\n [button] royal blue [button_]\nsize\n [button] small [button_]\n [button] medium [button_]\n [button] large [button_]\n [button] x-large [button_]\n [button] xx-large [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\nPrice: $19.99\nRating: N.A.\n[button] Description [button_]\n[button] Features [button_]\n[button] Reviews [button_]\n[button] Buy Now [button_]\n', Action 4: 'click[royal blue]']\n[Observation 5: 'You have clicked royal blue.\nYou have clicked men.\nInstruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\n[button] < Prev [button_]\nfit type\n [clicked button] men [clicked button_]\n [button] women [button_]\ncolor\n [button] black [button_]\n [button] navy [button_]\n [button] dark heather [button_]\n [button] neon pink [button_]\n [clicked button] royal blue [clicked button_]\nsize\n [button] small [button_]\n [button] medium [button_]\n [button] large [button_]\n [button] x-large [button_]\n [button] xx-large [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\nPrice: $19.99\nRating: N.A.\n[button] Description [button_]\n[button] Features [button_]\n[button] Reviews [button_]\n[button] Buy Now [button_]\n', Action 5: 'click[large]']
You are now at step 6 and your current observation is: You have clicked large.\nYou have clicked royal blue.\nYou have clicked men.\nInstruction:\nFind me machine wash men's t-shirts & tanks with polyester heathers, heathers cotton, cotton heather, needle sleeve, classic fit with color: royal blue, and fit type: men, and size: large, and price lower than 40.00 dollars\n[button] Back to Search [button_]\n[button] < Prev [button_]\nfit type\n [clicked button] men [clicked button_]\n [button] women [button_]\ncolor\n [button] black [button_]\n [button] navy [button_]\n [button] dark heather [button_]\n [button] neon pink [button_]\n [clicked button] royal blue [clicked button_]\nsize\n [button] small [button_]\n [button] medium [button_]\n [clicked button] large [clicked button_]\n [button] x-large [button_]\n [button] xx-large [button_]\nOnce Upon A Time Queen Born In 1982 T Shirts It Was Me Tank Top\nPrice: $19.99\nRating: N.A.\n[button] Description [button_]\n[button] Features [button_]\n[button] Reviews [button_]\n[button] Buy Now [button_]\n.
Your admissible actions of the current situation are: ['click[back to search]', 'click[< prev]', 'click[description]', 'click[features]', 'click[reviews]', 'click[buy now]', 'click[men]', 'click[women]', 'click[black]', 'click[navy]', 'click[dark heather]', 'click[neon pink]', 'click[royal blue]', 'click[small]', 'click[medium]', 'click[large]', 'click[x-large]', 'click[xx-large]'].

Now it's your turn to take one action for the current step.
You should first reason step-by-step about the current situation, then think carefully which admissible action best advances the shopping goal. This reasoning process MUST be enclosed within <think> </think> tags.
Once you've finished your reasoning, you should choose an admissible action for current step and present it within <action> </action> tags.

Response (WebShop, Step 6)

<think>\nTo find the desired items, I need to ensure that the color is royal blue, the fit type is classic, the size is large, and the price is lower than 40.00 dollars. The items should also have heathers, heathers cotton, cotton heather, needle sleeve, and classic fit. I should filter the results based on these criteria.\n</think>\n<action>click[buy now]</action>