FESGPO: FREE-ENERGY AND STRUCTURE-AWARE CREDIT ASSIGNMENT FOR MULTI-TURN AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement Learning (RL) for multi-turn Large Language Model (LLM) agents often needs to identify which intermediate decisions actually drive task completion from sparse terminal feedback. Existing group-based methods do not fully exploit the probabilities and costs of successful paths across rollouts or the decision structure within trajectories. To bridge this gap, we propose FESGPO, which treats the rollout group of a given task as a finite observation of the policy–environment interaction process. In this view, transition frequency and path length capture the reliability and cost of an action's route to success. From these statistics, FESGPO estimates empirical transition probabilities and aggregates all successful paths in the empirical graph into a statistical-physics free energy. The corresponding Bellman recursion decomposes this free energy into the shortest success distance, the empirical success probability, and the excess path cost. Furthermore, exact repeated transitions, temporal order, and the final outcome enable more precise attribution of terminal credit to specific decisions. FESGPO exploits this structure to redistribute terminal credit, concentrating the training signal on decisions that materially shape subsequent behavior. At Qwen2.5-1.5B, FESGPO improves strict success over the strongest trained baselines by 7.29 percentage points on ALFWorld and 2.87 percentage points on WebShop. It raises nonzero PPO-advantage coverage from 24.07% to 56.68% and surpasses GraphGPO with dynamic sampling on ALFWorld using 16.15% as many rollouts. We verify our main theoretical results in Lean 4. Our implementation is available at `https://anonymous.4open.science/r/FESGPO`.

1 INTRODUCTION

Large Language Model (LLM)-driven agents are increasingly deployed in real-world environments requiring sustained interaction (Wang et al., 2023; Zhou et al., 2024), where multi-turn decision-making is key to autonomous problem solving. In such settings, an action's impact often surfaces only after multiple subsequent steps, while complex goals require interdependent decisions to be formed, executed, and revised over time. Reinforcement Learning (RL) offers a natural paradigm for such learning, enabling policies to improve directly from environmental outcomes (Bai et al., 2024; Qi et al., 2025). However, the environment typically provides only one terminal outcome per trajectory, whereas policy optimization must assign updates to every constituent decision (Tan et al., 2026). As the horizon lengthens, more decisions fall under the same outcome, making their differential contributions increasingly difficult to discern (Zhang, 2026). Credit assignment thus becomes a central challenge, where training algorithms must convert sparse trajectory outcomes into learning signals that distinguish individual actions (Tao et al., 2026; Wang et al., 2026b).

However, existing group-based RL methods construct action-level credit from limited rollout statistics, trajectory-level terminal outcomes, or both, leading to two credit mismatches. At the cross-rollout level, transition reliability is often reduced to a statistic, such as shortest successful distance, average return, or empirical success rate. Such summaries overlook rollout information, including the number of successful paths and downstream path costs, causing structurally different transitions to receive similar credit and making robust actions difficult to distinguish from fragile ones (Cheng et al., 2026; Li et al., 2026; Zhong et al., 2025; Feng et al., 2025). At the within-trajectory level, terminal outcomes are propagated across actions in a trajectory without accounting for their structural roles. In successful trajectories, repeated or loop-related actions may therefore receive positive

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

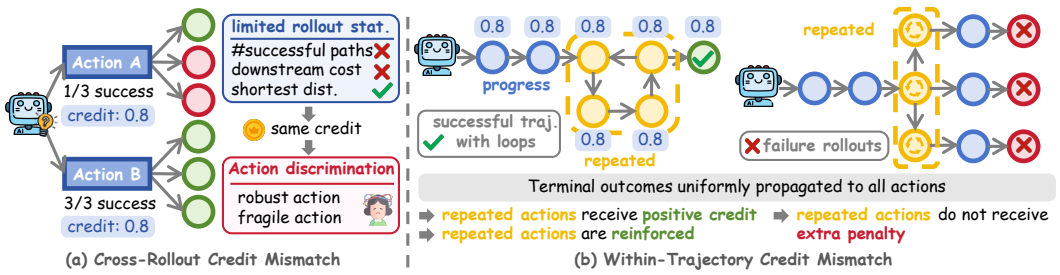

Figure 1: Credit assignment mismatches in multi-turn agents: (a) limited rollout statistics blur action reliability; (b) uniform terminal-outcome propagation obscures step contributions.

credit and be reinforced despite not advancing the task. In all-failure groups, the shared failure outcome provides no additional penalty for repeated actions, leaving redundant behaviors insufficiently distinguished (Xu et al., 2026; Zhang et al., 2026; Mesnard et al., 2021).

To address these gaps, we propose FESGPO (Free-Energy Group Policy Optimization with Structure-Aware Credit Refinement). It views each rollout group as a finite empirical trajectory ensemble and uses path statistics to estimate action prospects. The first stage evaluates actions from their empirical downstream path distributions. Actions with the same shortest successful distance can differ substantially in how often and how efficiently they reach success. Inspired by the partition function in statistical physics (Kivimäki et al., 2014), we weight successful paths by empirical transition probabilities and exponentially decayed path costs, and define free energy as the temperature-scaled negative log of the resulting success partition function. This yields a relative graph advantage that captures both success frequency and path cost.

The second stage refines terminal credit using within-trajectory state visitation structure. FESGPO detects exact repeated transitions and adjusts their credit according to trajectory order and group outcomes: it penalizes redundant repetitions in all-failure groups, while in mixed-outcome groups it transfers positive credit from repeated steps to subsequent decisions that escape the repeated region and clears negative episode credit at matching failed-trajectory transitions. This refinement preserves the graph advantage while reallocating trajectory-level credit. Together, the two stages produce step-level advantages from the same on-policy rollouts, requiring no additional models or environment interactions. Our contributions are as follows:

- We identify two credit-assignment mismatches in multi-turn Agent RL: shortest-distance valuation omits success support and costs of alternative successful paths, while trajectory-level outcomes obscure the contribution of individual decisions.
- We propose FESGPO, a two-stage framework that combines free-energy credit over empirical successful paths with structure-aware refinement of within-trajectory outcome credit, using the same on-policy rollouts without additional models or environment interactions.
- FESGPO outperforms the compared agentic RL baselines on ALFWorld and WebShop using Qwen2.5-Instruct at both 1.5B and 7B scales. At 1.5B, it improves ALFWorld and WebShop strict success rates by 7.29 and 2.87 percentage points, respectively, while surpassing all compared 7B agentic RL baselines and outperforming GraphGPO with dynamic sampling on ALFWorld using 16.15% as many rollouts.

2 Related Work

2.1 Multi-turn Agent Reinforcement Learning

Multi-turn agent tasks are widely used to evaluate LLM agents (Shridhar et al., 2021; Yao et al., 2022). GRPO extends trajectory-level outcome optimization to multi-turn interaction (Shao et al., 2024), but provides limited step-level distinction. Recent methods improve credit estimation through anchor state-level grouping in GiGPO (Feng et al., 2025), difficulty-aware modulation in ProxMO (Fang et al., 2026), and hierarchical grouping in HGPO (He et al., 2026a). However, these methods

still rely on trajectory-level outcomes as comparison signals, causing relative advantages to collapse when all rollouts fail. ProGPO addresses this with first-visit state coverage (Yang et al., 2026), while AVSPO injects virtual rewards (He et al., 2026b). GraphGPO instead aggregates rollouts into a state-transition graph and estimates values using global shortest-path statistics (Cheng et al., 2026). Despite improving outcome-based feedback, these methods still provide limited step-level attribution of which intermediate transitions drive task progress.

2.2 Credit Assignment in Agent Reinforcement learning

Credit assignment in multi-turn agent RL has two directions: forward evaluation of whether a transition leads to success, and backward attribution of credit to the steps that produced it. In the forward direction, HCAPO uses outcome-conditioned hindsight critics for step-level value estimation (Tan et al., 2026), while GraphGPO and G2PO estimate success prospects through shortest-path and multi-level graph statistics (Cheng et al., 2026; Wang et al., 2026a). However, these methods collapse the future into a single statistic, obscuring path distribution and failure risk, thus failing to judge how stably a transition leads to success. In the backward direction, GRPO broadcasts trajectory-level rewards uniformly across all intermediate steps (Shao et al., 2024), making it hard to distinguish effective decisions from redundant behaviors. To address this, RSPO aligns step-level contributions through reward swapping (Liu et al., 2026), SPA-RL constructs step-wise progress attribution (Wang et al., 2025), and RLVMR introduces verifiable process rewards to reduce repetitive behaviors (Zhang et al., 2026). However, these methods do not fully exploit the decision structure within individual trajectories. In particular, loop-closing actions may receive positive credit merely by co-occurring with success, while transitions that actually drive the trajectory out of loops may not.

3 Preliminaries

3.1 Multi-turn Agentic Tasks

Let $x \sim p(X)$ denote a task sampled from the task distribution, and let π_θ denote a language-model policy parameterized by θ. At step t, the policy observes state s_t, samples an action $a_t \in \mathcal{V}^n$ from $\pi_\theta(a_t \mid s_t, x)$, and receives the next state s_{t+1} from the environment. Here, $\mathcal{V}$ is the token vocabulary and n is the maximum generation length of one action. A complete interaction forms a trajectory $\tau = \{(s_0, a_0), (s_1, a_1), \ldots, (s_{T-1}, a_{T-1})\}$, where T is the trajectory length. Upon termination, the environment returns a scalar outcome $R(\tau)$. Multi-turn agentic tasks typically provide sparse terminal feedback, leaving most intermediate steps without direct environment rewards.

3.2 Group-normalized Graph Credit

GraphGPO (Cheng et al., 2026) aggregates trajectories sampled for the same task into a directed state-transition graph $G = (S, E)$, where identical states are merged into a single node. Each observed transition (s, a, s') becomes a directed edge $(s, a, s', c(s, a)) \in E$, with transition cost $c(s, a) > 0$, typically set to 1. Successful terminal observations map to a designated success state s_{succ}. Task progress is measured by the minimum path cost from each state to s_{succ}:

$$d(s) = \begin{cases} 0, & s = s_{\text{succ}}, \\ \min_{(s,a,s',c)\in E}\{c + d(s')\}, & s \rightsquigarrow s_{\text{succ}}, \\ +\infty, & \text{otherwise}, \end{cases} \tag{1}$$

where $s \rightsquigarrow s_{\text{succ}}$ means s can reach s_{succ} in the graph. A smaller $d(s)$ indicates that the state is closer to task completion. For states that cannot reach s_{succ}, the infinite distance is replaced by $d_{\max} + 1$, where $d_{\max}$ is the largest finite $d(s)$ in the graph.

The unit-cost implementation uses the successor score $R^G(s, a, s') = r_{\text{succ}}\, \omega^{d(s')}$, where $r_{\text{succ}} > 0$ is the success reward and $\omega \in (0, 1)$ is the distance discount factor. Transition occurrences from the same source state form groups $G^G(s)$, retaining repeated observations. Graph rewards are standardized within each group:

$$A^G(s, a, s') = \frac{R^G(s, a, s') - \mu(G^G(s))}{\sigma(G^G(s)) + \varepsilon}, \tag{2}$$

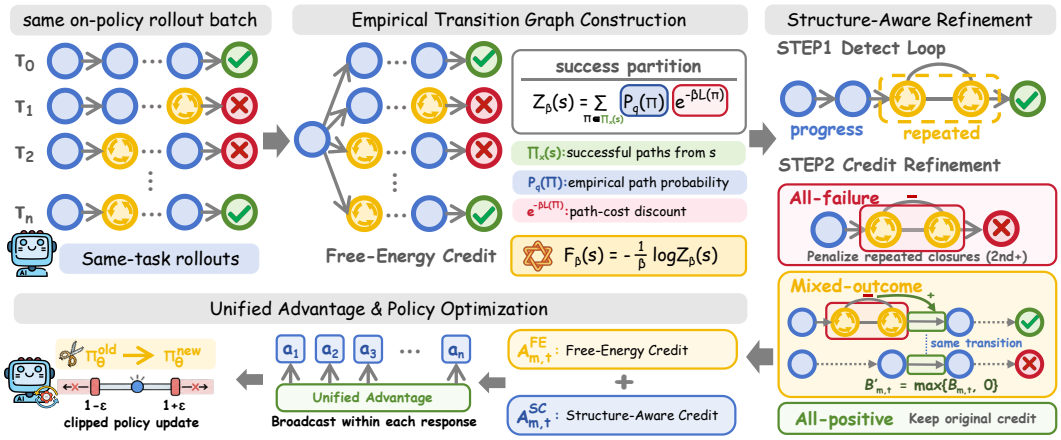

Figure 2: Overview of FESGPO. Free-energy credit evaluates transition reliability from successful-path distributions, while structure-aware refinement adjusts credit according to repeated closures and outcome groups. The two signals form a unified advantage for clipped policy optimization.

where μ and σ are the mean and standard deviation of the graph rewards in $G^G(s)$, respectively, and $\varepsilon > 0$ stabilizes normalization. Singleton groups receive zero advantage.

For each transition in trajectory τ, the graph-level credit is combined with the GRPO episode-level advantage $A^E(\tau)$ as $A(s, a, s') = \beta^G A^G(s, a, s') + \beta^E A^E(\tau)$, where β^G and β^E control the weights of the graph and trajectory advantages, respectively.

4 Methodology

As shown in Figure 2, FESGPO constructs informative step-level credit from the same on-policy rollout batch through two complementary stages. Free-energy credit aggregates empirical successful-path probabilities and costs to evaluate downstream transition reliability, while structure-aware refinement adjusts episode credit using repeated closures, persistent exits, and rollout outcomes. The resulting signals are then combined into a unified advantage for clipped policy optimization without any additional environment interactions.

4.1 Free-Energy Credit

Free-energy credit evaluates each successor by aggregating the probability and cost of its successful continuations in the empirical rollout graph, assigning higher credit to transitions more strongly supported by efficient paths to task completion.

For task x, we merge identical states across its rollout group and count transition occurrences as $N_x(s, a, s')$. We define $N_x^{\text{out}}(s) = \sum_{a,s'} N_x(s, a, s')$ and the empirical kernel $q_x(a, s' \mid s) = N_x(s, a, s')/N_x^{\text{out}}(s)$ for states with $N_x^{\text{out}}(s) > 0$. The denominator counts all outgoing transition occurrences, so branches unable to reach success lower the empirical success probability. Non-success states with no observed outgoing transitions have $q_x = 0$ and terminate without success. We map successful terminal observations to the common absorbing node $s^\dagger$, where paths end.

The empirical graph combines observed transitions into successful continuations, including those spanning different rollouts through shared states. Let $\Pi_x(s)$ contain all finite supported paths from s to $s^\dagger$, allowing repeated states before success. For a path $\pi = ((s_t, a_t, s_{t+1}))_{t=0}^{n-1}$, define $P_{q_x}(\pi) = \prod_{t=0}^{n-1} q_x(a_t, s_{t+1} \mid s_t)$ and $L(\pi) = \sum_{t=0}^{n-1} c(s_t, a_t)$, with $c(s, a) > 0$. Following the path-partition formulation of Kivimäki et al. (2014), we aggregate these paths using $\beta = -\log \gamma > 0$:

$$Z_\beta(s) = \sum_{\pi \in \Pi_x(s)} P_{q_x}(\pi) e^{-\beta L(\pi)}, \qquad F_\beta(s) = -\frac{1}{\beta} \log Z_\beta(s). \tag{3}$$

Here $\gamma \in (0, 1)$ controls how path weights decay with cost; experiments use $c(s, a) = 1$. The empty path gives $Z_\beta(s^\dagger) = 1$ and $F_\beta(s^\dagger) = 0$. States without successful continuations have $Z_\beta = 0$ and $F_\beta = +\infty$.

A Bellman recursion computes this aggregate without enumerating paths. Let $\mathcal{T}_x = \{s_1, \ldots, s_n\}$ contain the success-reachable nonterminal states. Conditioning on the first transition gives

$$Z_\beta(s) = \sum_{a,s'} q_x(a, s' \mid s) e^{-\beta c(s,a)} Z_\beta(s'), \qquad s \in \mathcal{T}_x. \tag{4}$$

With $z_i = Z_\beta(s_i)$, this yields the sparse system $(I - W_\beta)\mathbf{z} = \mathbf{b}_\beta$, where $[W_\beta]_{ij} = \sum_a q_x(a, s_j \mid s_i) e^{-\beta c(s_i,a)}$ and $[\mathbf{b}_\beta]_i = \sum_a q_x(a, s^\dagger \mid s_i) e^{-\beta c(s_i,a)}$. We solve this system with the original q_x and stated boundary values. Positive costs make the Bellman map contractive on the finite graph, giving a unique nonnegative solution and finite free energies on success-reachable states (Appendix C.1).

Free energy expresses the path aggregate in cost units and separates three contributions. Let $d_{\mathrm{SP}}(s) = \min_{\pi \in \Pi_x(s)} L(\pi)$, $p_+^{(q)}(s) = \sum_{\pi \in \Pi_x(s)} P_{q_x}(\pi)$, and $\eta_\beta(s)$ be the conditional expectation of $e^{-\beta(L - d_{\mathrm{SP}}(s))}$ given success under q_x.

Proposition 1 (Free-energy decomposition)**.** *For every success-reachable state s,*

$$F_\beta(s) = d_{\mathrm{SP}}(s) - \frac{\log p_+^{(q)}(s)}{\beta} - \frac{\log \eta_\beta(s)}{\beta} \geq d_{\mathrm{SP}}(s). \tag{5}$$

The two nonnegative corrections account for unsuccessful continuations and excess cost along successful paths. For example, if every successful path from two successors costs d, success probabilities 1 and $1/3$ yield partition values γ^d and $\gamma^d/3$. Their shortest distances coincide, but their path aggregates preserve the difference in empirical success probability. Appendix C.2 proves the decomposition; Appendix C.3 shows $F_\beta \to d_{\mathrm{SP}}$ as $\beta \to \infty$ with the empirical graph fixed.

We standardize successor scores across transition occurrences sharing a source state. Let $\mathcal{S}_x^+ = \mathcal{T}_x \cup \{s^\dagger\}$ and $F_{\max} = \max_{s \in \mathcal{S}_x^+} F_\beta(s)$. For training, set $\bar{F}_\beta(s) = F_\beta(s)$ on $\mathcal{S}_x^+$ and $\bar{F}_\beta(s) = F_{\max} + 1$ elsewhere. This assigns unreachable successors a common score lower than that of any reachable successor. With success reward $r_{\mathrm{succ}} > 0$, the raw score and training credit are

$$r^{\mathrm{FE}}(s, a, s') = r_{\mathrm{succ}} \gamma^{\bar{F}_\beta(s')}, \qquad A_{m,t}^{\mathrm{FE}} = \frac{\widehat{r}_{m,t}^{\mathrm{FE}} - \widehat{\mu}_{\mathrm{FE}}(s_{m,t})}{\widehat{\sigma}_{\mathrm{FE}}(s_{m,t}) + \varepsilon}. \tag{6}$$

Here $\widehat{r}^{\mathrm{FE}}$ includes the inherited invalid-action penalty; $\widehat{\mu}_{\mathrm{FE}}$ and $\widehat{\sigma}_{\mathrm{FE}}$ are its mean and sample standard deviation over outgoing transition occurrences, with $\varepsilon > 0$ (Appendix C.4). Singleton groups receive zero advantage. For reachable successors, $r^{\mathrm{FE}} = r_{\mathrm{succ}} Z_\beta(s')$; success transitions receive r_{succ}. Without observed success, raw scores coincide; transitions sharing a source state receive zero FE advantage when their invalid-action penalties match, motivating the structural failure signal in Section 4.2.

4.2 Structure-Aware Credit Refinement

Free-energy credit evaluates successful continuations in the empirical graph, while episode credit broadcasts an outcome-derived signal across a trajectory. Structure-aware refinement distinguishes repeated closures from persistent exits to allocate this episode credit.

We track visited states in order. Transition $e_{m,t} = (s_{m,t}, a_{m,t}, s_{m,t+1})$ closes a loop when its successor revisits this sequence; $U_{m,t}$ contains states from that occurrence to the source. We then remove states after the revisited state. Let $N_m^{\mathrm{cl}}(e; t)$ count closures of transition e through step t in trajectory m, with $N_m^{\mathrm{cl}}(e_{m,t}; t) \geq 2$ indicating repetition. A persistent exit is a later transition leaving $U_{m,t}$ without returning in the observed suffix, and $u_{m,t}^\star$ denotes its position. Appendix D.1 details validity, matching, and exit construction.

In all-failure groups, repeated closures supply additional local negative evidence without requiring an exit. Let $M_x^{\mathrm{rep}}(e)$ count trajectories with at least two valid closures of e, including the current

trajectory. For each repeated closure, apply

$$\delta_{m,t} = N_m^{\mathrm{cl}}(e_{m,t}; t) + M_x^{\mathrm{rep}}(e_{m,t}) - 2, \quad \Delta_{m,t}^{\mathrm{fail}} = -\rho_x \frac{\delta_{m,t}}{\delta_{m,t}+1}. \tag{7}$$

The correction is zero elsewhere. Here ρ_x is the root mean square of nonzero free-energy advantages at valid positions within task x, defaulting to 1 for an empty set. Repetition within and across trajectories increases the penalty magnitude, bounded by ρ_x (Appendix D.3).

In mixed groups, we transfer positive episode credit from repeated closures to persistent exits within positive-outcome trajectories. Let $\ell_{m,t} > 0$ denote the number of unmasked tokens and $B_{m,t} = \ell_{m,t} A^E(\tau_m)$ the corresponding episode credit mass at valid positions. Let $\mathcal{D}_m^+$ contain repeated closures with $B_{m,t} > 0$ and a persistent exit. Credit is redistributed simultaneously, with each donor contributing its original mass once and multiple donors allowed to share an exit, preserving total mass within each positive trajectory:

$$B'_{m,u} = B_{m,u} - \mathbf{1}[u \in \mathcal{D}_m^+] B_{m,u} + \sum_{t \in \mathcal{D}_m^+,\, u^\star_{m,t} = u} B_{m,t}. \tag{8}$$

A transition that serves as a persistent exit in a positive trajectory may also appear in a failed trajectory. We therefore reuse this cross-trajectory evidence to remove conflicting negative episode credit. Let $\mathcal{R}_x$ collect exit transitions paired with repeated closures in positive trajectories. At valid positions in non-positive trajectories with $e_{m,t} \in \mathcal{R}_x$, we set $B'_{m,t} = \max\{B_{m,t}, 0\}$; other positions retain their episode credit, and free-energy credit is unchanged. We then recover $\widetilde{A}^E_{m,t} = B'_{m,t}/\ell_{m,t}$. Outside mixed groups, episode credit remains unchanged, with no adjustment for all-positive groups. The structural component is $A^{\mathrm{SC}}_{m,t} = \widetilde{A}^E_{m,t} + \Delta^{\mathrm{fail}}_{m,t}$.

4.3 Unified Advantage and Policy Optimization

Free-energy credit captures downstream success prospects in the empirical graph, while structure-aware credit refines transitions using trajectory structure. We combine the two components into a unified transition-level advantage, $A^{\mathrm{FESGPO}}_{m,t} = \lambda_{\mathrm{FE}} A^{\mathrm{FE}}_{m,t} + \lambda_{\mathrm{SC}} A^{\mathrm{SC}}_{m,t}$, which is shared by all unmasked response tokens associated with transition (m,t). We then optimize the policy with the token-level clipped PPO surrogate

$$\ell_{m,t,k}(\theta) = \min\Big\{ r_{m,t,k}(\theta) A^{\mathrm{FESGPO}}_{m,t}, \mathrm{clip}\Big(r_{m,t,k}(\theta), 1-\epsilon^-_{\mathrm{clip}}, 1+\epsilon^+_{\mathrm{clip}}\Big) A^{\mathrm{FESGPO}}_{m,t} \Big\}. \tag{9}$$

The core maximization objective is

$$\mathcal{J}_{\mathrm{clip}}(\theta) = \mathbb{E}_{\mathcal{B}} \left[\frac{1}{N_{\mathcal{B}}} \sum_{(m,t,k)\in\mathcal{B}} \chi_{m,t,k}\, \ell_{m,t,k}(\theta) \right] - \alpha_{\mathrm{KL}} D_{\mathrm{KL}}(\pi_\theta \,\|\, \pi_{\mathrm{ref}}). \tag{10}$$

Here, $\mathcal{B}$ is a microbatch, $\chi_{m,t,k}$ the response-token mask, $N_{\mathcal{B}} = \sum \chi_{m,t,k}$, and $r_{m,t,k}(\theta)$ the current-to-old token probability ratio. Advantages and old-policy probabilities are fixed during updates; Appendix E details dual clipping, entropy regularization, and the sampled KL estimator.

5 Experiments

We evaluate FESGPO through four questions: **RQ1:** Does FESGPO improve the performance frontier across different model scales? **RQ2:** Do the two credit-assignment stages provide complementary gains? **RQ3:** How does the path discount affect the free-energy credit signal? **RQ4:** How much useful training signal can FESGPO extract from a fixed rollout budget?

5.1 Experiment Setup

Benchmarks. We evaluate on ALFWorld (Shridhar et al., 2021) and WebShop (Yao et al., 2022). ALFWorld contains six task types (Pick, Clean, Cool, Look, Heat, and Pick Two); we report strict task success. WebShop reports both its native Task Score and strict success, which measure graded task completion and exact fulfillment, respectively.

Table 1: Main results on ALFWorld and WebShop. We report ALFWorld strict success (%) and WebShop Task Score / strict success (%). Results are mean (sample standard deviation) over three seeds when available; bold denotes the best trained-model result within each scale.

Type	Method	ALFWorld							WebShop	
		Pick	Clean	Cool	Look	Heat	Pick2	All	Score	Success
Closed-Source Models										
Prompting	GPT-4o	73.53(3.03)	15.48(5.54)	18.03(7.32)	57.65(22.09)	44.43(3.76)	36.52(11.79)	41.00(3.12)	13.91(5.95)	6.00(3.60)
Prompting	GPT-5.6-sol	96.72(2.25)	73.13(7.04)	88.93(12.09)	83.70(9.50)	83.78(5.63)	74.15(7.56)	84.50(1.73)	46.23(5.85)	29.33(4.73)
Prompting	Gemini-3.5-Flash	95.40(1.11)	64.97(11.38)	84.29(6.32)	39.46(6.71)	71.15(6.03)	86.13(11.57)	78.33(4.00)	39.45(1.26)	26.67(0.58)
Open-Weight Models										
Prompting	Kimi-K3	91.37(4.32)	59.74(5.16)	72.45(4.63)	56.00(1.71)	16.93(7.32)	65.21(2.96)	65.33(3.88)	26.23(2.05)	16.00(1.00)
Prompting	GLM-5.2	88.22(4.92)	89.34(6.45)	89.49(8.27)	75.88(18.63)	63.38(1.97)	77.85(3.28)	82.33(3.40)	56.09(2.80)	32.67(1.53)
Qwen2.5-1.5B-Instruct										
Prompting	Qwen2.5	10.52(4.74)	2.31(2.01)	1.33(2.31)	0.00(0.00)	0.00(0.00)	3.17(5.50)	3.91(0.00)	25.80(4.18)	10.16(3.91)
Prompting	ReAct	5.31(3.78)	0.00(0.00)	0.00(0.00)	0.00(0.00)	0.00(0.00)	0.00(0.00)	1.30(0.91)	22.27(3.14)	8.33(1.63)
Prompting	Reflexion	11.56(1.63)	5.69(5.06)	0.00(0.00)	3.33(5.77)	2.78(4.81)	0.00(0.00)	4.69(1.57)	39.78(2.49)	15.89(2.51)
RL Training	PPO	64.80(3.50)	40.50(6.90)	57.10(4.90)	60.60(6.60)	46.40(4.00)	47.40(1.90)	54.40(3.10)	73.80(3.00)	51.50(2.90)
RL Training	RLOO	88.30(3.00)	52.80(8.60)	71.00(5.90)	62.80(8.70)	66.40(5.50)	56.90(4.70)	69.70(2.50)	73.90(5.60)	52.10(6.70)
RL Training	GRPO	85.85(0.23)	69.88(2.83)	67.05(5.49)	68.92(10.84)	70.97(2.03)	66.41(11.62)	71.61(3.85)	84.99(1.78)	74.74(3.85)
RL Training	GiGPO	98.89(1.92)	93.30(2.64)	79.26(7.49)	75.64(1.11)	**100.00**(0.00)	86.70(3.69)	90.10(0.90)	86.36(1.15)	75.52(1.80)
RL Training	GraphGPO	93.02(2.40)	98.67(2.31)	84.29(9.92)	96.30(6.42)	**100.00**(0.00)	84.81(4.32)	91.93(4.44)	86.99(2.12)	77.34(2.34)
RL Training	HGPO	97.26(2.46)	97.73(1.97)	81.38(12.37)	90.20(10.60)	90.95(3.47)	89.57(5.62)	91.15(1.97)	86.94(2.37)	75.26(6.07)
RL Training	BEACON	93.00(2.70)	98.48(2.62)	83.51(6.06)	91.67(8.33)	**100.00**(0.00)	87.52(6.69)	91.15(1.63)	86.42(1.93)	76.30(1.80)
RL Training	**FESGPO**	**100.00**(0.00)	**100.00**(0.00)	**100.00**(0.00)	**100.00**(0.00)	94.59(4.79)	**97.62**(4.12)	**99.22**(0.78)	**90.27**(2.07)	**80.21**(3.93)
Qwen2.5-7B-Instruct										
Prompting	Qwen2.5	42.10(4.69)	15.50(9.98)	12.28(8.41)	24.39(5.93)	7.41(1.60)	19.09(9.97)	22.14(4.00)	42.52(2.69)	14.58(3.69)
Prompting	ReAct	29.44(3.15)	36.16(10.58)	17.67(6.66)	33.79(7.83)	9.26(8.49)	1.59(2.74)	21.35(3.53)	36.28(4.13)	13.02(2.74)
Prompting	Reflexion	47.48(10.30)	41.47(6.58)	19.00(9.54)	43.48(17.59)	11.11(4.81)	2.92(2.56)	28.91(0.78)	56.69(1.74)	25.78(2.34)
RL Training	PPO	92.30(4.00)	64.00(8.40)	92.50(2.40)	89.50(7.00)	80.30(2.00)	68.80(8.30)	80.40(2.70)	81.40(3.10)	68.70(5.10)
RL Training	RLOO	87.60(4.30)	78.20(8.30)	87.30(5.80)	81.30(7.60)	71.90(5.20)	48.90(8.40)	75.50(4.60)	80.30(3.20)	65.70(4.00)
RL Training	GRPO	93.44(6.92)	88.58(4.44)	70.54(8.55)	78.50(1.54)	94.44(5.56)	73.63(7.96)	84.38(2.34)	83.53(1.19)	75.00(3.13)
RL Training	GiGPO	99.07(1.60)	94.75(1.96)	82.94(1.30)	91.37(10.28)	95.24(4.12)	95.71(4.17)	93.49(1.80)	86.97(1.48)	78.12(3.58)
RL Training	GraphGPO	99.02(1.70)	97.15(2.48)	91.43(6.63)	93.33(11.55)	94.44(9.62)	94.96(5.63)	95.31(2.71)	88.84(1.09)	78.91(2.82)
RL Training	HGPO	**100.00**(0.00)	94.61(1.79)	81.00(7.25)	**100.00**(0.00)	95.24(8.25)	94.61(6.09)	92.97(3.12)	88.86(2.10)	78.39(1.97)
RL Training	BEACON	94.61(1.71)	92.39(0.43)	73.98(3.35)	97.44(4.44)	76.59(6.11)	90.25(4.24)	88.54(1.19)	87.87(3.56)	79.69(3.91)
RL Training	**FESGPO**	**100.00**(0.00)	**100.00**(0.00)	**100.00**(0.00)	97.78(3.85)	**100.00**(0.00)	**97.33**(4.62)	**99.22**(0.78)	**89.20**(1.32)	**80.99**(1.97)

Models and baselines. We train Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct policies. The comparison includes prompting with GPT-4o, GPT-5.6-sol, Gemini-3.5-Flash, Kimi-K3, and GLM-5.2, Qwen2.5 agents with ReAct and Reflexion, and trained agents using PPO, RLOO, GRPO (Shao et al., 2024), GiGPO (Feng et al., 2025), GraphGPO (Cheng et al., 2026), HGPO (He et al., 2026a), and BEACON (Wang et al., 2026b). GraphGPO provides the matched graph-credit baseline; the remaining methods cover trajectory-level, state-grouped, history-grouped, and milestone-based credit.

Training protocol. Each FESGPO update uses 16 groups of 8 on-policy trajectories (128 total) for both stages. Runs share the optimizer, prompt and response length limits, maximum horizon, evaluation schedule, and task split. Tables 1–2 report our trained methods' final Step-150 test results as mean and sample standard deviation over three evaluation seeds (128 episodes per seed). We use $\gamma = 0.2$ on ALFWorld and $\gamma = 0.1$ on WebShop; see Appendix F for details.

5.2 Main Results

Table 1 compares prompting and Agent-RL methods across model scales to evaluate whether FESGPO advances the performance frontier at different scales. At 1.5B, FESGPO achieves 99.22% success on ALFWorld, improving over the strongest trained baseline by 7.29 percentage points and reaching 100% on four task types. On WebShop, it achieves a Task Score of 90.27 and 80.21% strict success, improving by 3.28 and 2.87 percentage points, respectively. At 7B, FESGPO likewise achieves 99.22% success on ALFWorld and 80.99% strict success on WebShop, outperforming the strongest baselines by 3.91 and 1.30 percentage points, respectively. Notably, 1.5B FESGPO outperforms all 7B Agent-RL baselines on both ALFWorld and WebShop strict success, indicating that its gains extend beyond simply increasing model scale and consistently improve Agent-RL performance at both evaluated scales, thereby advancing the model-scale–performance frontier.

5.3 Ablation Study

Table 2 compares FESGPO variants at 1.5B. The shortest-path variant retains graph credit and structure-aware refinement, replacing free-energy valuation with GraphGPO's shortest-path valuation. ALFWorld success is 94.53%, versus 99.22% for the full model. The variant without structure-aware refinement retains free-energy credit and the original episode advantage, achieving 94.01% ALFWorld success. Within this stage, removing the mixed-outcome and all-failure branches causes drops of 3.65 and 3.13 percentage points, respectively, showing that the former improves credit redistribution in mixed-outcome trajectories while the latter recovers structural signals from all-failure rollouts. The full model achieves the highest mean on all three metrics, indicating that cross-path success structure and within-trajectory structural refinement jointly improve credit assignment.

Using the frozen Step 150 ALFWorld 1.5B policy, we compare independent continuations from matched decision contexts. Across 36 contexts where candidate successors have equal shortest success distances but different FE values, FE-preferred actions achieve 2.43 percentage points higher mean success than the alternative. Taking a persistent-exit action yields 33.68 points higher mean success than letting the policy choose the initial action (nine interventions across five tasks). These findings are consistent with the ablation results, supporting FE-based action ranking and credit allocation to persistent exits. Appendix B provides the full protocol and results.

Table 2: Component ablation of FESGPO on Qwen2.5-1.5B-Instruct. All variants use the same training budget; results are mean (sample standard deviation) over three seeds.

Configuration	ALFWorld success	WebShop score	WebShop success
w/ shortest-path valuation	94.53 (0.78)	90.08 (0.29)	78.13 (3.58)
w/o structure-aware refinement	94.01 (1.19)	90.12 (3.28)	77.60 (4.30)
w/o mixed-outcome refinement	95.57 (0.90)	90.16 (1.29)	79.17 (4.01)
w/o all-failure refinement	96.09 (0.78)	90.18 (1.56)	79.95 (3.25)
FESGPO (full)	**99.22 (0.78)**	**90.27 (2.07)**	**80.21 (3.93)**

5.4 Parameter Sensitivity Analysis

To examine how the path discount affects free-energy credit, we vary γ while keeping the optimizer and rollout protocol fixed. Figure 3 reports benchmark metrics on a four-point grid.

Since $\beta = -\log\gamma$, smaller γ causes path weights to decay more rapidly with increasing cost, placing stronger emphasis on low-cost paths, whereas larger γ weakens this cost preference and assigns relatively more weight to higher-cost alternatives. Both benchmarks achieve their best performance at interior values of the search range: ALFWorld peaks at $\gamma = 0.2$, while WebShop obtains its highest Task Score and strict success rate at $\gamma = 0.1$. Performance decreases toward both extremes, suggesting that neither aggressively suppressing higher-cost paths nor treating path cost too weakly provides reliable credit signals. These results support the role of γ as a trade-off parameter: effective free-energy credit assignment requires balancing empirical support from frequently observed paths against the cost associated with reaching future outcomes.

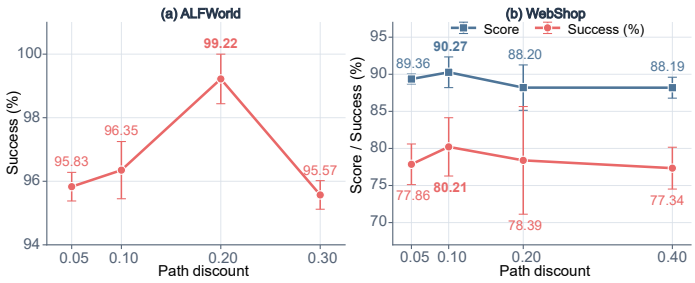

Figure 3: Path-discount sensitivity: test results at Step 150.

5.5 Efficiency Analysis

FESGPO increases credit coverage without changing rollout generation or the policy loss. Figure 4 reports the training rollout success rates of Qwen2.5-7B-Instruct on ALFWorld and WebShop. All methods are compared over the same 0–150 optimization-step budget.

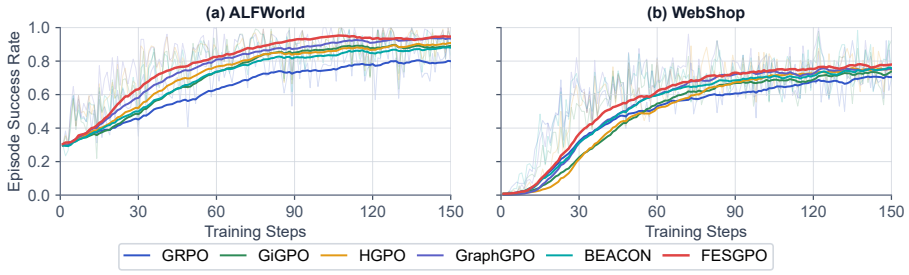

Figure 4: Training rollout success rates of Qwen2.5-7B-Instruct on ALFWorld (left) and WebShop (right) across training steps, measured over the rollouts collected at each step.

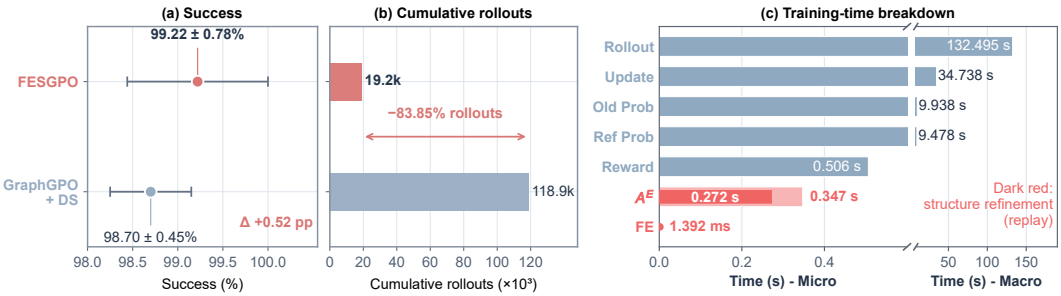

Figure 5: Sampling efficiency and training-time breakdown for Qwen2.5-1.5B-Instruct on ALFWorld. (a) Step-150 success. (b) Cumulative environment rollouts. (c) Mean per-step training time, with structure refinement and free-energy solve (FE) shown separately.

On ALFWorld with Qwen2.5-1.5B-Instruct, we compare FESGPO with GraphGPO using dynamic sampling (DS; Appendix F.3) over 150 optimization steps. FESGPO achieves 99.22% success with 19,200 rollouts (128 per step), versus 98.70% with 118,912 (792.75 per step) for GraphGPO+DS, using 16.15% as many rollouts (Figure 5). At Step 95, cumulative costs are 12,160 rollouts and 12.360 GPU-hours for FESGPO, versus 52,480 rollouts and 39.871 GPU-hours for GraphGPO+DS.

The matched GraphGPO and FESGPO runs differ by 1.394 s per optimization step (0.75%). Figure 5 shows that the full FESGPO advantage stage accounts for 0.185% of step time, while the sparse free-energy solve accounts for 0.00074%. A cross-method timing sweep gives 237.062 s for GRPO, 278.449 s for GiGPO, 219.294 s for HGPO, 186.786 s for GraphGPO, and 188.180 s for FESGPO. The added credit computation remains a small part of training while increasing the fraction of action positions with nonzero PPO advantage from 24.07% to 56.68%.

6 Conclusion

We introduced FESGPO, a two-stage method for multi-turn agentic RL that derives transition-level credit from the same on-policy rollout batch. Free-energy credit aggregates empirical successful-path probabilities and continuation costs, while structure-aware refinement adjusts episode credit using repeated closures, persistent exits, and rollout outcomes. Together, these stages capture cross-rollout success prospects and within-trajectory structural contributions without additional models or environment interactions. With both Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct, FESGPO achieves the highest strict success rates among the evaluated agentic RL methods on ALFWorld and WebShop. The 1.5B model achieves 99.22% and 80.21% strict success, improving over the strongest same-scale baselines by 7.29 and 2.87 percentage points, respectively, while surpassing all evaluated 7B agentic RL baselines. The 7B model achieves 99.22% and 80.99%, exceeding the strongest same-scale baselines by 3.91 and 1.30 percentage points, respectively. Ablations support the complementary contributions of both stages. On ALFWorld with the 1.5B model, FESGPO also increases nonzero PPO-advantage coverage from 24.07% to 56.68%. It outperforms GraphGPO with dynamic sampling using only 16.15% as many rollouts.

AI USE STATEMENT

We used LLMs as writing assistants to polish the manuscript, correcting grammatical errors, improving stylistic consistency, and refining phrasing for clarity and conciseness.

ETHICS STATEMENT

This work studies credit assignment for multi-turn language-model agents using the public ALFWorld and WebShop benchmarks. The experiments do not involve human participants or the collection of new personal data. Applications in real-world environments should include appropriate safety evaluation, limits on tool permissions, and human oversight.

REPRODUCIBILITY STATEMENT

We provide pseudocode in Appendix A, mechanism-analysis protocols in Appendix B, and theoretical assumptions and proofs in Appendices C and D. Appendix C.5 describes the Lean 4 formalization. Appendices E and F specify policy optimization and implementation details, including prompts, graph-state representations, and training configurations. Section 5.1 describes the checkpoint and evaluation protocol. Our implementation is available in our anonymous repository.

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. doi: 10.18653/v1/2024.acl-long.662. URL https://aclanthology.org/2024.acl-long.662/.

Hao Bai, Yifei Zhou, Mert Cemri, Jiayi Pan, Alane Suhr, Sergey Levine, and Aviral Kumar. Digirl: Training in-the-wild device-control agents with autonomous reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 12461–12495, 2024.

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. *arXiv preprint arXiv:2605.26684*, 2026.

Yangyi Fang, Jiaye Lin, Xiaoliang Fu, Cong Qin, Haolin Shi, and Chang Liu. Proximity-based multi-turn optimization: Practical credit assignment for LLM agent training. In Yunyao Li, Georg Rehm, and Mei Tu (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pp. 285–307, San Diego, California, USA, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-394-4. doi: 10.18653/v1/2026.acl-industry.19. URL https://aclanthology.org/2026.acl-industry.19/.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. Group-in-group policy optimization for LLM agent training. In *Advances in Neural Information Processing Systems*, volume 38, pp. 46375–46408, 2025.

Shuo He, Lang Feng, Qi Wei, Xin Cheng, Lei Feng, and Bo An. Hierarchy-of-groups policy optimization for long-horizon agentic tasks. In *International Conference on Learning Representations*, volume 2026, pp. 27572–27593, 2026a.

Xixiang He, Qiyao Sun, Ao Cheng, Xingming Li, Xuanyu Ji, Hailun Lu, Runke Huang, and Qingyong Hu. Advantage collapse in group relative policy optimization: Diagnosis and mitigation. In *Forty-third International Conference on Machine Learning*, 2026b. URL https://openreview.net/forum?id=MKNimf9bIx.

Ilkka Kivimäki, Masashi Shimbo, and Marco Saerens. Developments in the theory of randomized shortest paths with a comparison of graph node distances. *Physica A: Statistical Mechanics and its Applications*, 393:600–616, 2014.

Yuanfan Li, Qi Zhou, Wenjing Duan, and Lu Chen. When denser credit is not enough: Evidence-calibrated policy optimization for long-horizon LLM agent training. *arXiv preprint arXiv:2606.05885*, 2026.

Qiang Liu, Taian Guo, Ruizhi Qiao, and Xing Sun. RSPO: Reward-swap policy optimization for multi-turn LLM agents. *arXiv preprint arXiv:2607.04713*, 2026.

Thomas Mesnard, Theophane Weber, Fabio Viola, Shantanu Thakoor, Alaa Saade, Anna Harutyunyan, Will Dabney, Thomas S Stepleton, Nicolas Heess, Arthur Guez, Eric Moulines, Marcus Hutter, Lars Buesing, and Remi Munos. Counterfactual credit assignment in model-free reinforcement learning. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 7654–7664. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/mesnard21a.html.

Zehan Qi, Xiao Liu, Iat Long Iong, Hanyu Lai, Xueqiao Sun, Jiadai Sun, Xinyue Yang, Yu Yang, Shuntian Yao, Wei Xu, Jie Tang, and Yuxiao Dong. WebRL: Training LLM web agents via self-evolving online curriculum reinforcement learning. In *International Conference on Learning Representations*, volume 2025, pp. 79791–79821, 2025.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. URL https://arxiv.org/abs/1707.06347.

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 *Advances in Neural Information Processing Systems*, 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 *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. URL https://arxiv.org/abs/2010.03768.

Hui-Ze Tan, Xiao-Wen Yang, Hao Chen, Jie-Jing Shao, Yi Wen, Yuteng Shen, Weihong Luo, Xiku Du, Lan-Zhe Guo, and Yu-Feng Li. Hindsight credit assignment for long-horizon LLM agents, 2026. URL https://arxiv.org/abs/2603.08754.

Leitian Tao, Baolin Peng, Wenlin Yao, Tao Ge, Hao Cheng, Mike Hang Wang, Jianfeng Gao, and Sharon Li. TRACE: Turn-level reward assignment via credit estimation for long-horizon agents, 2026. URL https://arxiv.org/abs/2607.13988.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. In *Intrinsically-Motivated and Open-Ended Learning Workshop @NeurIPS2023*, 2023. URL https://openreview.net/forum?id=nfx5IutEed.

Hanlin Wang, Chak Tou Leong, Jiashuo Wang, Jian Wang, and Wenjie Li. SPA-RL: Reinforcing LLM agents via stepwise progress attribution. *arXiv preprint arXiv:2505.20732*, 2025.

Yunan Wang, Minghui Song, Zihan Zhang, Shaohan Huang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, and Qi Zhang. Group-graph policy optimization for long-horizon agentic reinforcement learning, 2026a. URL https://arxiv.org/abs/2606.22995.

Zixuan Wang, Yuchen Yan, Hongxing Li, Teng Pan, Dingming Li, Ruiqing Zhang, Weiming Lu, Jun Xiao, Yueting Zhuang, and Yongliang Shen. Milestone-guided policy learning for long-horizon language agents, 2026b. URL https://arxiv.org/abs/2605.06078.

Tianshi Xu, Yuteng Chen, and Meng Li. CLEANER: Self-purified trajectories boost agentic reinforcement learning. In *The 1st Workshop on Scaling Post-training for LLMs*, 2026. URL https://openreview.net/forum?id=bWPhJpPlnc.

Kaibing Yang, Guangfeng Cai, Shengtian Yang, Shuo He, Yu Li, Mengyi Liu, Pengwei Chen, Jun Xu, and Lei Feng. Progress-conditioned group policy optimization for long-horizon agentic tasks. *arXiv preprint arXiv:2607.22724*, 2026.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. WebShop: Towards scalable real-world web interaction with grounded language agents. In *Advances in Neural Information Processing Systems*, 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 *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Chenchen Zhang. From reasoning to agentic: Credit assignment in reinforcement learning for large language models, 2026. URL https://arxiv.org/abs/2604.09459.

Zijing Zhang, Ziyang Chen, Mingxiao Li, Zhaopeng Tu, and Xiaolong Li. RLVMR: Reinforcement learning with verifiable meta-reasoning rewards for robust long-horizon agents. In *International Conference on Learning Representations*, volume 2026, pp. 114676–114700, 2026.

Shan Zhong, Shutong Ding, He Diao, Xiangyu Wang, Kah Chan Teh, and Bei Peng. Flow-Critic: Bridging value estimation with flow matching in reinforcement learning. *arXiv preprint arXiv:2510.22686*, 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 *International Conference on Learning Representations*, volume 2024, pp. 15585–15606, 2024.

A PSEUDOCODE

Algorithm 1 computes both credit components from the same rollout batch. The group branch uses raw task outcomes, independent of invalid-action penalties. Structural refinement leaves all-positive groups unchanged and preserves token mass within each positive trajectory of mixed groups.

Algorithm 1 The training procedure of FESGPO

Require: Initial policy π_θ, fixed reference policy π_{ref}, task distribution $p(X)$,
group size M, maximum steps T, path discount $\gamma \in (0,1)$,
normalization tolerance ε, PPO settings in Appendix E

$\beta \leftarrow -\log\gamma$; $\lambda_{\mathrm{FE}} \leftarrow 1$; $\lambda_{\mathrm{SC}} \leftarrow 1$
for each training iteration **do**
  $\theta_{\mathrm{old}} \leftarrow \theta$; sample a task batch $\mathcal{X}$ from $p(X)$
  for all tasks $x \in \mathcal{X}$ **do**
    Sample M trajectories $\{\tau_m\}_{m=1}^{M}$ with π_{old}, each with at most T steps
    Retain raw task outcomes $R(\tau_m)$, executed actions, validity sets $\mathcal{A}_m$, and masks χ
    // Free-energy credit
    Build the unit-cost task graph and occurrence kernel q_x; map successful terminals to $s^\dagger$
    Set $Z_\beta(s^\dagger) = 1$ and $Z_\beta = 0$ on success-unreachable states
    Solve $(I - W_\beta)\mathbf{z} = \mathbf{b}_\beta$ on $\mathcal{T}_x$ using Eq. (16)
    Recover F_β and its finite extension $\bar{F}_\beta$ (Appendix C.4)
    Compute $r^{\mathrm{FE}}_{m,t} = r_{\mathrm{succ}}\gamma^{\bar{F}_\beta(s_{m,t+1})}$
    Apply the invalid-action penalty to graph and token rewards before normalization
    Compute source-normalized A^{FE} and the original episode component $A^{E,0}$
    // Structure-aware credit refinement
    Identify eligible repeated closures $\mathcal{E}_m$ and footprints $U_{m,t}$ (Appendix D.1)
    Define $\ell_{m,t} = \sum_k \chi_{m,t,k} > 0$ and $B_{m,t} = \sum_k \chi_{m,t,k} A^{E,0}_{m,t,k}$
    $B' \leftarrow B$; $\Delta^{\mathrm{fail}} \leftarrow 0$; $H_x \leftarrow \sum_m \mathbf{1}[R(\tau_m) > 0]$

if $H_x = 0$ **then**
  Compute ρ_x from valid nonzero A^{FE}, with empty-set value 1 (Appendix D.3)
  for all m and $t \in \mathcal{E}_m$ **do**
    $\delta_{m,t} \leftarrow N_m^{\mathrm{cl}}(e_{m,t}; t) + M_x^{\mathrm{rep}}(e_{m,t}) - 2$
    $\Delta_{m,t}^{\mathrm{fail}} \leftarrow -\rho_x \delta_{m,t}/(\delta_{m,t}+1)$
  end for
else if $0 < H_x < M$ **then**
  Find persistent exits $u_{m,t}^{\star}$ for repeated closures in positive trajectories
  Form witness sets $\mathcal{J}_m$ and exit-transition set $\mathcal{R}_x$ (Appendix D.1)
  $\mathcal{D}_m^{+} \leftarrow \{t \in \mathcal{J}_m : B_{m,t} > 0\}$
  Transfer each donor's original mass to its exit simultaneously using Eq. (8)
  for all (m,t) with $R(\tau_m) \le 0$, $t \in \mathcal{A}_m$, and $e_{m,t} \in \mathcal{R}_x$ **do**
    $B'_{m,t} \leftarrow \max\{B_{m,t}, 0\}$
  end for
end if
$\widetilde{A}_{m,t}^{E} \leftarrow B'_{m,t}/\ell_{m,t}$; $A_{m,t}^{\mathrm{SC}} \leftarrow \widetilde{A}_{m,t}^{E} + \Delta_{m,t}^{\mathrm{fail}}$
$A_{m,t}^{\mathrm{FESGPO}} \leftarrow \lambda_{\mathrm{FE}} A_{m,t}^{\mathrm{FE}} + \lambda_{\mathrm{SC}} A_{m,t}^{\mathrm{SC}}$
Broadcast $A_{m,t}^{\mathrm{FESGPO}}$ to the unmasked response tokens
end for
Update θ on the collected batch using the clipped policy loss in Eq. (68)
end for

B Mechanism Analysis

B.1 Experimental protocol

We use the frozen Qwen2.5-1.5B-Instruct Step 150 policy in the enhanced ALFWorld environment from Appendix F.4. Sampling uses temperature 1.0, a two-step history, and a total budget of 50 environment steps, with prompt and response limits of 2,048 and 512 tokens. For each of 140 in-distribution tasks, we collect eight discovery rollouts and construct an empirical graph following Section 4.1, with $\gamma = 0.2$. We select contexts from these rollouts in a fixed task-balanced order before collecting continuation outcomes. The graph and policy remain fixed during evaluation.

We restore each context by resetting the environment and replaying its recorded action prefix, checking observations, admissible actions, and the policy input. Each comparison uses the same remaining episode budget and 32 continuations per arm, with paired randomization across arms. After the initial action, both arms follow the frozen policy. We count strict task success, treating failures and timeouts as zero, and average the success-rate differences across intervention records. We compute 95% confidence intervals from 10,000 task-cluster bootstrap samples, resampling both arms jointly.

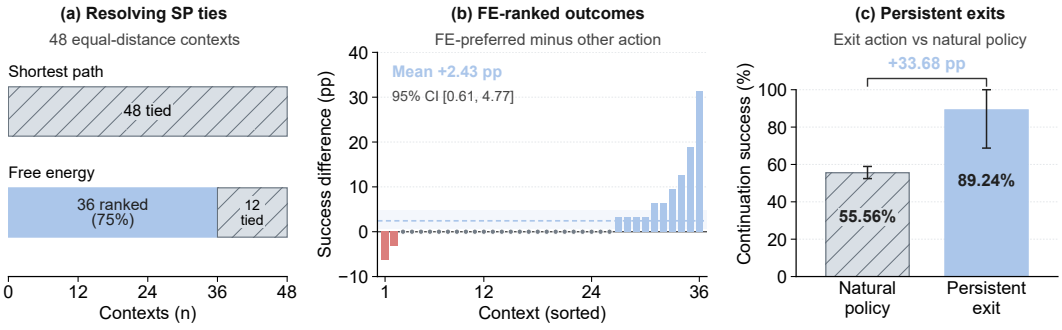

Figure 6: Action-level mechanism analysis on ALFWorld at Step 150. (a) FE discrimination among equal-shortest-distance contexts. (b) Continuation-success differences between the FE-preferred action and the other candidate, including all 36 FE-distinguished contexts. (c) Persistent-exit actions versus the natural-policy control. Each arm uses 32 continuations. The shaded band in (b) and error bars in (c) show 95% task-bootstrap confidence intervals.

B.2 FE RESOLVES SHORTEST-PATH TIES

We compare two legal candidate actions at the same decision context. Both valuations score their successors on the same discovery graph using unit transition costs and the success boundary from Section 4.1. We use the raw values before invalid-action penalties and source-state normalization; lower free energy determines the FE-preferred action.

Among 48 selected contexts with equal shortest success distances, FE distinguishes 36 and retains ties in 12 (Figure 6a). The directional comparison includes all 36 FE-distinguished contexts, covering 36 tasks. Their successor observations match the discovery graph. Independent continuations yield 10 positive differences, 24 ties, and two negative differences in success rate (Figure 6b). In 21 of the tied contexts, both candidates succeed in all 32 continuations. Table 3 reports the aggregate success rates and their difference. FE's additional ranking identifies actions with higher mean continuation success within this equal-distance panel.

B.3 PERSISTENT EXITS IMPROVE CONTINUATION SUCCESS

We examine discovery rollouts from all tasks in the ALFWorld `valid_seen` split and identify persistent exits in successful trajectories from mixed-outcome groups following Section 4.2 and Appendix D. We identify 26 eligible donor-exit pairs across five tasks, each transferring positive episode credit from a repeated closure to a persistent exit. Following a predetermined task-balanced rule, we select at most two pairs per task, yielding nine intervention records covering these five tasks. We complete selection before continuation evaluation.

At each original exit context, one arm executes the recorded exit action and then follows the frozen policy. The natural-policy arm samples both its initial action and subsequent actions. We retain natural-policy continuations that choose the same initial action, as well as continuations that return to the loop region, fail, or time out.

The comparison contains nine intervention records across eight distinct contexts and five tasks. One context occurs in two donor-exit pairs, each with independent continuation samples. Seven interventions favor the exit action, one ties, and one favors the natural-policy control (Figure 6c). The aggregate results support allocating credit to persistent exits at these contexts. The FE and exit comparisons use different controls and measure local action behavior; their differences do not decompose training-level gains.

Table 3: Independent continuation success on ALFWorld with the frozen Step 150 policy. Target and control denote the first- and second-listed conditions, respectively.

	Success (%)		Target − control (pp)	
Comparison	Target	Control	Difference	95% CI
FE-preferred vs. other candidate	88.98	86.55	**+2.43**	[0.61, 4.77]
Persistent exit vs. natural policy	89.24	55.56	**+33.68**	[12.15, 47.19]

FE: 36 records (36 contexts, 36 tasks); 1,025/1,152 vs. 997/1,152 successes.
Exit: 9 records (8 contexts, 5 tasks); 257/288 vs. 160/288 successes.
Each record uses 32 continuations per arm. Confidence intervals use paired task-cluster bootstrap resampling.

B.4 CREDIT-SIGNAL DIAGNOSTICS

We audit all 11,408 transition occurrences in the same 1,120 discovery rollouts by recomputing credit with the training implementation. The 140 groups comprise 128 all-success, 11 mixed-outcome, and one all-failure group. All 58 eligible repeated-closing occurrences receive negative all-failure corrections, covering 4,740 response tokens; their original graph and episode advantages are zero. No correction falls outside the eligibility gate. The mixed branch records 26 donor-exit transfers, including eight recipient positions with multiple donors, and removes negative episode credit at 17 matching positions in failed trajectories. Across successful trajectories, the maximum absolute change in total episode token-credit mass is only 4.995×10^{-5}.

C FREE-ENERGY CREDIT: DEFINITIONS AND PROOFS

The success partition has a unique Bellman solution, and its negative logarithm separates shortest-path cost from success probability and excess cost. We establish these claims for a fixed task x and its finite empirical rollout graph. Every observed transition has a finite cost $c(s, a) > 0$. Throughout the existence and decomposition arguments, we fix $\beta = -\log \gamma > 0$, with $\gamma \in (0, 1)$. All path probabilities refer to the empirical kernel q_x.

C.1 EXISTENCE AND UNIQUENESS OF THE FREE-ENERGY SOLUTION

Empirical process and stopping rule. Let $\mathcal{S}_x$ contain the observed states and the common success state $s^\dagger$. Within task x, graph construction merges occurrences with the same state representation across sampled trajectories. The transition counts retain each occurrence's multiplicity. We map successful terminal observations to $s^\dagger$ and stop the process upon reaching it. For $s \neq s^\dagger$, let $N_x(s, a, s')$ count transition occurrences and define

$$\begin{aligned} N_x^{\mathrm{out}}(s) &= \sum_{a,s'} N_x(s, a, s'), \\ q_x(a, s' \mid s) &= \begin{cases} N_x(s, a, s')/N_x^{\mathrm{out}}(s), & N_x^{\mathrm{out}}(s) > 0, \\ 0, & N_x^{\mathrm{out}}(s) = 0. \end{cases} \end{aligned} \tag{11}$$

The denominator counts every observed outgoing transition, including transitions caused by invalid actions and those from unsuccessful trajectories. For each state with $N_x^{\mathrm{out}}(s) > 0$, the probabilities $q_x(a, s' \mid s)$ sum to one over all (a, s'). A non-success state with no observed outgoing transitions terminates the empirical process without success. This stopping convention uses the observed graph without adding a failure state. We assign the success boundary value directly, so the recursion does not use outgoing transitions from $s^\dagger$.

Finite successful paths. Let $\Pi_x^{(n)}(s)$ contain supported successful paths of n transitions. Each such path satisfies

$$\begin{aligned} \pi &= ((s_0, a_0, s_1), \ldots, (s_{n-1}, a_{n-1}, s_n)), \\ s_0 &= s, \qquad s_n = s^\dagger, \qquad s_t \neq s^\dagger \quad (0 \le t < n), \\ q_x(a_t, s_{t+1} \mid s_t) &> 0 \quad (0 \le t < n). \end{aligned} \tag{12}$$

For $n = 0$, only $s = s^\dagger$ admits a path: the empty path. We assign it probability one and cost zero. For all other finite paths, define

$$P_{q_x}(\pi) = \prod_{t=0}^{n-1} q_x(a_t, s_{t+1} \mid s_t), \qquad L(\pi) = \sum_{t=0}^{n-1} c(s_t, a_t). \tag{13}$$

The complete path set is $\Pi_x(s) = \bigcup_{n=0}^{\infty} \Pi_x^{(n)}(s)$. It allows repeated states before success and any finite number of cycle traversals. A supported path may combine transitions from different sampled trajectories because the empirical process follows q_x at each state.

Distinct successful paths specify disjoint events of this stopped process. Their total probability is the eventual success probability under q_x:

$$\begin{aligned} p_+^{(q)}(s) &= \sum_{\pi \in \Pi_x(s)} P_{q_x}(\pi) \le 1, \\ 0 \le Z_\beta(s) &= \sum_{\pi \in \Pi_x(s)} P_{q_x}(\pi) e^{-\beta L(\pi)} \le p_+^{(q)}(s). \end{aligned} \tag{14}$$

Thus the path sum converges, even when the graph contains cycles. The probability $p_+^{(q)}(s)$ describes the empirical process and need not equal the rollout success fraction. The empty path gives $Z_\beta(s^\dagger) = 1$. A state without a successful path has an empty path set and $Z_\beta(s) = 0$.

First-step decomposition and the linear system. Let $\mathcal{T}_x = \{s_1, \ldots, s_n\}$ contain the success-reachable states other than $s^\dagger$. Every path from $s \in \mathcal{T}_x$ begins with one observed transition and then follows a successful continuation. Splitting the nonnegative path sum by this first transition gives the recursion in Eq. (4):

$$Z_\beta(s) = \sum_{a,s'} q_x(a, s' \mid s) e^{-\beta c(s,a)} Z_\beta(s'), \qquad s \in \mathcal{T}_x. \tag{15}$$

Success contributes its boundary value one, and success-unreachable successors contribute zero. With $z_i = Z_\beta(s_i)$, define

$$\begin{aligned} [W_\beta]_{ij} &= \sum_a q_x(a, s_j \mid s_i) e^{-\beta c(s_i,a)}, \\ [\mathbf{b}_\beta]_i &= \sum_a q_x(a, s^\dagger \mid s_i) e^{-\beta c(s_i,a)}, \\ (I - W_\beta)\mathbf{z} &= \mathbf{b}_\beta. \end{aligned} \tag{16}$$

We retain the original q_x denominators when restricting the solve to $\mathcal{T}_x$. Transitions to unreachable states keep their probability mass in the kernel and contribute zero continuation value. Dropping their zero contributions from the linear system does not renormalize the surviving transitions.

Proposition 2 (Existence and uniqueness of the Bellman solution)**.** *For a finite empirical graph with finite positive transition costs and $\beta > 0$, the system in Eq. (16) has a unique nonnegative solution. Its entries equal the sums over successful paths in Eq. (3). Every success-reachable state has a strictly positive partition value and a uniquely defined finite free energy.*

Proof. **Empty system.** If $\mathcal{T}_x = \emptyset$, the only success-reachable state is $s^\dagger$. The boundary values specify $Z_\beta(s^\dagger) = 1$ and zero at every other state, so no linear solve is necessary. The statements about entries of $\mathbf{z}$ then concern the empty vector. We assume $\mathcal{T}_x \neq \emptyset$ for the remaining steps.

Contraction bound. Every state in $\mathcal{T}_x$ has an observed outgoing transition. The finite transition support and positive costs give

$$c_{\min} = \min_{\substack{s \in \mathcal{T}_x,\ a, s': \\ q_x(a,s'|s) > 0}} c(s,a) > 0, \qquad \kappa_\beta = e^{-\beta c_{\min}} \in (0,1). \tag{17}$$

For every row i, nonnegativity and the original kernel normalization imply

$$\begin{aligned} \sum_{j=1}^n [W_\beta]_{ij} &= \sum_{j=1}^n \sum_a q_x(a, s_j \mid s_i) e^{-\beta c(s_i,a)} \\ &\leq \kappa_\beta \sum_{j=1}^n \sum_a q_x(a, s_j \mid s_i) \\ &\leq \kappa_\beta \sum_{a,s'} q_x(a, s' \mid s_i) = \kappa_\beta < 1. \end{aligned} \tag{18}$$

Hence $\|W_\beta\|_\infty \leq \kappa_\beta < 1$. This bound places every eigenvalue strictly inside the unit disk. It also makes the affine Bellman map $\mathbf{z} \mapsto W_\beta \mathbf{z} + \mathbf{b}_\beta$ a contraction in the infinity norm.

Existence and nonnegativity. Submultiplicativity gives $\|W_\beta^k\|_\infty \leq \kappa_\beta^k$, so the matrix series $S = \sum_{k=0}^\infty W_\beta^k$ converges. Its partial sums satisfy

$$(I - W_\beta) \sum_{k=0}^K W_\beta^k = \left(\sum_{k=0}^K W_\beta^k \right) (I - W_\beta) = I - W_\beta^{K+1} \longrightarrow I. \tag{19}$$

Taking the limit yields $S = (I - W_\beta)^{-1}$. Since both W_β and $\mathbf{b}_\beta$ are nonnegative, we obtain the nonnegative solution

$$\mathbf{z} = (I - W_\beta)^{-1} \mathbf{b}_\beta = \sum_{k=0}^\infty W_\beta^k \mathbf{b}_\beta \geq 0. \tag{20}$$

Term-by-term path interpretation. The entry $[\mathbf{b}_\beta]_i$ sums all one-transition paths from s_i to success. Each multiplication by W_β prepends one transition whose successor lies in $\mathcal{T}_x$. Expanding the products, or inducting on k, therefore gives

$$[W_\beta^k \mathbf{b}_\beta]_i = \sum_{\pi \in \Pi_x^{(k+1)}(s_i)} P_{q_x}(\pi) e^{-\beta L(\pi)}. \tag{21}$$

A successful path cannot pass through a success-unreachable state. It therefore stays in $\mathcal{T}_x$ until its final transition to $s^\dagger$, so the expansion includes every path of the indicated length. Summing over k recovers the complete partition:

$$z_i = \sum_{k=0}^{\infty} \sum_{\pi \in \Pi_x^{(k+1)}(s_i)} P_{q_x}(\pi) e^{-\beta L(\pi)} = Z_\beta(s_i). \tag{22}$$

Repeated state visits contribute terms at their corresponding finite path lengths.

Uniqueness. Invertibility already ensures uniqueness. Equivalently, any two solutions $\mathbf{z}$ and $\mathbf{z}'$ satisfy $\|\mathbf{z} - \mathbf{z}'\|_\infty = \|W_\beta(\mathbf{z} - \mathbf{z}')\|_\infty \le \kappa_\beta \|\mathbf{z} - \mathbf{z}'\|_\infty$. Since $\kappa_\beta < 1$, their difference is zero.

Strict positivity and finite free energy. For each $s \in \mathcal{T}_x$, choose one supported finite successful path π. Every factor in its probability is positive and its cost is finite. Consequently,

$$0 < P_{q_x}(\pi) e^{-\beta L(\pi)} \le Z_\beta(s) \le p_+^{(q)}(s) \le 1. \tag{23}$$

The unique positive partition value thus defines a unique finite negative logarithm. Including the boundary states, the complete free-energy definition is

$$F_\beta(s) = \begin{cases} 0, & s = s^\dagger, \\ -\beta^{-1} \log Z_\beta(s) \in [0, \infty), & s \in \mathcal{T}_x, \\ +\infty, & s \in \mathcal{S}_x \setminus (\mathcal{T}_x \cup \{s^\dagger\}). \end{cases} \tag{24}$$

These boundary values also cover the empty-system case. □

C.2 Proof of the free-energy decomposition

Shortest successful cost and the conditional distribution. Fix a success-reachable state s. Removing a repeated-state cycle preserves the remaining supported path and strictly reduces its cost. Repeating this operation leaves a simple path. The finite empirical graph has finitely many supported simple paths, so a shortest successful path exists. Define

$$d_{\mathrm{SP}}(s) = \min_{\pi \in \Pi_x(s)} L(\pi), \qquad \widetilde{P}_s(\pi) = \frac{P_{q_x}(\pi)}{p_+^{(q)}(s)}, \qquad \sum_{\pi \in \Pi_x(s)} \widetilde{P}_s(\pi) = 1. \tag{25}$$

Strict positivity of $p_+^{(q)}(s)$ makes $\widetilde{P}_s$ the conditional path distribution given success. Define the discounted excess-cost term

$$\begin{aligned} \eta_\beta(s) &= \sum_{\pi \in \Pi_x(s)} \widetilde{P}_s(\pi) e^{-\beta(L(\pi) - d_{\mathrm{SP}}(s))} \\ &= \mathbb{E}_{\widetilde{P}_s}\left[e^{-\beta(L - d_{\mathrm{SP}}(s))} \right]. \end{aligned} \tag{26}$$

Every path has $L(\pi) \ge d_{\mathrm{SP}}(s)$, so its exponential factor lies in $(0, 1]$. If π_* is one shortest successful path, its factor equals one. Hence

$$0 < \widetilde{P}_s(\pi_*) \le \eta_\beta(s) \le 1, \qquad 0 < p_+^{(q)}(s) \le 1. \tag{27}$$

Proof of Proposition 1. For each path, split its cost into the shortest successful cost and its excess. Extracting the common shortest-path discount gives

$$\begin{aligned} Z_\beta(s) &= \sum_{\pi \in \Pi_x(s)} P_{q_x}(\pi) e^{-\beta[d_{\mathrm{SP}}(s) + (L(\pi) - d_{\mathrm{SP}}(s))]} \\ &= e^{-\beta d_{\mathrm{SP}}(s)} \sum_{\pi \in \Pi_x(s)} P_{q_x}(\pi) e^{-\beta(L(\pi) - d_{\mathrm{SP}}(s))} \\ &= e^{-\beta d_{\mathrm{SP}}(s)} p_+^{(q)}(s) \eta_\beta(s). \end{aligned} \tag{28}$$

All three factors are strictly positive. Taking $-\beta^{-1}\log$ yields

$$F_\beta(s) = d_{\mathrm{SP}}(s) \quad \underbrace{-\frac{\log p_+^{(q)}(s)}{\beta}}_{\text{success-probability correction}} \quad \underbrace{-\frac{\log \eta_\beta(s)}{\beta}}_{\text{excess-cost correction}} \geq d_{\mathrm{SP}}(s). \tag{29}$$

The bounds above make both corrections nonnegative. The first correction vanishes if and only if $p_+^{(q)}(s) = 1$, meaning that the empirical process eventually reaches success with probability one.

For the second correction, write

$$1 - \eta_\beta(s) = \sum_{\pi \in \Pi_x(s)} \widetilde{P}_s(\pi)\left[1 - e^{-\beta(L(\pi) - d_{\mathrm{SP}}(s))}\right]. \tag{30}$$

Every summand is nonnegative. Their sum vanishes if and only if every positive-probability successful path has zero excess cost. We therefore obtain the equality condition

$$F_\beta(s) = d_{\mathrm{SP}}(s) \quad \Longleftrightarrow \quad \begin{cases} p_+^{(q)}(s) = 1, \\ L(\pi) = d_{\mathrm{SP}}(s) \text{ for every supported successful path } \pi. \end{cases} \tag{31}$$

At success, the empty path gives $d_{\mathrm{SP}} = 0$ and $p_+^{(q)} = \eta_\beta = 1$, so the same identity holds. Success-unreachable states use the boundary $F_\beta = +\infty$; the conditional path distribution applies only to success-reachable states. □

For two success-reachable states u, v with equal shortest-path cost, subtraction gives

$$F_\beta(u) - F_\beta(v) = -\frac{1}{\beta}\log\left(\frac{p_+^{(q)}(u)\eta_\beta(u)}{p_+^{(q)}(v)\eta_\beta(v)}\right). \tag{32}$$

Thus free energy retains differences in successful-path probability and excess cost when shortest-distance scores coincide.

Schematic equal-distance example. Consider two states u, v whose supported successful paths all have the same cost $d > 0$. Set $p_+^{(q)}(u) = 1$ and $p_+^{(q)}(v) = 1/3$; the remaining probability at v leads to an unreachable branch. These assumptions give $d_{\mathrm{SP}}(u) = d_{\mathrm{SP}}(v) = d$ and $\eta_\beta(u) = \eta_\beta(v) = 1$. Equation (28) then yields

$$\begin{aligned} Z_\beta(u) &= \gamma^d, \qquad Z_\beta(v) = \frac{\gamma^d}{3}, \\ F_\beta(u) &= d, \qquad F_\beta(v) = d + \frac{\log 3}{\beta}, \\ F_\beta(v) - F_\beta(u) &= \frac{\log 3}{\beta}. \end{aligned} \tag{33}$$

The free-energy gap isolates the success-probability correction at a fixed successful-path cost.

C.3 Shortest-distance limit

We hold the empirical graph, its transition costs, and q_x fixed while varying β. For a success-reachable state s, define the probability mass of shortest successful paths:

$$p_*(s) = \sum_{\substack{\pi \in \Pi_x(s): \\ L(\pi) = d_{\mathrm{SP}}(s)}} P_{q_x}(\pi), \qquad 0 < p_*(s) \leq p_+^{(q)}(s) \leq 1. \tag{34}$$

A shortest supported path exists, so $p_*(s) > 0$. Its value is independent of β. Keeping only shortest paths gives a lower bound on the partition. Bounding every path discount by $e^{-\beta d_{\mathrm{SP}}(s)}$ gives an upper bound. Therefore,

$$\begin{aligned} p_*(s)e^{-\beta d_{\mathrm{SP}}(s)} \leq Z_\beta(s) \leq p_+^{(q)}(s)e^{-\beta d_{\mathrm{SP}}(s)} \leq e^{-\beta d_{\mathrm{SP}}(s)}, \\ d_{\mathrm{SP}}(s) \leq F_\beta(s) \leq d_{\mathrm{SP}}(s) - \frac{\log p_*(s)}{\beta}. \end{aligned} \tag{35}$$

Since $-\log p_*(s)$ is a finite constant, the upper and lower bounds converge to the same value:

$$\lim_{\beta\to\infty} F_\beta(s) = d_{\mathrm{SP}}(s). \tag{36}$$

This is a limit of the free-energy distance. At finite β, equality holds under two conditions proved in Appendix C.2. Then, the unadjusted reward for a transition with successor s also equals $r_{\mathrm{succ}}\gamma^{d_{\mathrm{SP}}(s)}$.

C.4 Finite extension and transition credit

Finite values for training rewards. The theoretical solution keeps $Z_\beta = 0$ and $F_\beta = +\infty$ on success-unreachable states. For reward construction, define

$$\begin{aligned} \mathcal{S}_x^+ &= \mathcal{T}_x \cup \{s^\dagger\}, \qquad & F_{\max} &= \max_{s\in\mathcal{S}_x^+} F_\beta(s), \\ \bar{F}_\beta(s) &= \begin{cases} F_\beta(s), & s \in \mathcal{S}_x^+, \\ F_{\max}+1, & s \notin \mathcal{S}_x^+. \end{cases} \end{aligned} \tag{37}$$

The set $\mathcal{S}_x^+$ contains $s^\dagger$, and all its free energies are finite. Its maximum exists even when no observed state can reach success. With $r_{\mathrm{succ}} > 0$ and $\beta = -\log\gamma$, the raw transition score is

$$r^{\mathrm{FE}}(s,a,s') = r_{\mathrm{succ}}\gamma^{\bar{F}_\beta(s')} = \begin{cases} r_{\mathrm{succ}} Z_\beta(s'), & s' \in \mathcal{S}_x^+, \\ r_{\mathrm{succ}}\gamma^{F_{\max}+1}, & s' \notin \mathcal{S}_x^+. \end{cases} \tag{38}$$

Because $0 < \gamma < 1$, unreachable successors receive a common raw score strictly below every reachable successor's raw score. The finite extension changes only reward construction; the original partition values remain zero on unreachable states.

For transitions e_u, e_v with success-reachable successors u, v, Eq. (28) gives

$$\frac{r^{\mathrm{FE}}(e_u)}{r^{\mathrm{FE}}(e_v)} = \gamma^{d_{\mathrm{SP}}(u)-d_{\mathrm{SP}}(v)} \frac{p_+^{(q)}(u)\eta_\beta(u)}{p_+^{(q)}(v)\eta_\beta(v)}. \tag{39}$$

This ratio combines shortest-path cost with the probability and excess cost of successful continuations. The probability and excess-cost terms can outweigh a difference in shortest distance. In the equal-distance example above, the ratio of the raw scores is three.

Unit costs and successor scores. The experiments use unit transition costs, $c(s,a) = 1$, and $r_{\mathrm{succ}} = 10$. Equation (6) scores the continuation from the successor s'. Its partition value accounts for all costs from s' onward, and a transition directly to success receives raw score r_{succ}. We use this successor score, without multiplying it by an additional γ for the current action's unit cost. The implemented shortest-path variant uses the same successor-only convention, replacing $\bar{F}_\beta(s')$ with the correspondingly extended $d_{\mathrm{SP}}(s')$. Both variants apply the same invalid-action penalty and normalization after scoring.

If the rollout group contains no successful terminal, no observed path reaches $s^\dagger$. Then $\mathcal{S}_x^+ = \{s^\dagger\}$ and $F_{\max} = 0$, so every observed successor receives the same raw score $r_{\mathrm{succ}}\gamma$.

Invalid-action adjustment and source normalization. The inherited invalid-action penalty adjusts scores before normalization. Appendix D.1 defines the valid-action positions $\mathcal{A}_m$. For coefficient $\alpha_{\mathrm{inv}} \geq 0$, write

$$\widehat{r}^{\mathrm{FE}}_{m,t} = r^{\mathrm{FE}}(e_{m,t}) - \alpha_{\mathrm{inv}}\mathbf{1}[t \notin \mathcal{A}_m]. \tag{40}$$

Let $\mathcal{O}_x(s) = \{(m,t) : s_{m,t} = s\}$ collect outgoing transition occurrences from source state s within task x, and let $n_s = |\mathcal{O}_x(s)|$. Repeated observations retain their multiplicity. For each nonempty source group, we recompute the mean and sample standard deviation from the adjusted scores:

$$\begin{aligned} \widehat{\mu}_{\mathrm{FE}}(s) &= \frac{1}{n_s}\sum_{(m,t)\in\mathcal{O}_x(s)} \widehat{r}^{\mathrm{FE}}_{m,t}, \\ \widehat{\sigma}_{\mathrm{FE}}(s) &= \sqrt{\frac{1}{n_s-1}\sum_{(m,t)\in\mathcal{O}_x(s)} \left(\widehat{r}^{\mathrm{FE}}_{m,t} - \widehat{\mu}_{\mathrm{FE}}(s)\right)^2}, \qquad n_s > 1. \end{aligned} \tag{41}$$

The normalized training component uses these adjusted statistics in Eq. (6):

$$A^{\mathrm{FE}}_{m,t} = \begin{cases} \dfrac{\widehat{r}^{\mathrm{FE}}_{m,t} - \widehat{\mu}_{\mathrm{FE}}(s_{m,t})}{\widehat{\sigma}_{\mathrm{FE}}(s_{m,t}) + \varepsilon}, & n_{s_{m,t}} > 1, \\ 0, & n_{s_{m,t}} = 1, \end{cases} \qquad \varepsilon > 0. \tag{42}$$

Equal adjusted scores yield zero centered credit, and the positive ε keeps the denominator nonzero. When $\alpha_{\mathrm{inv}} = 0$, this operation reduces to normalization of the raw free-energy scores.

For a rollout group with no successful terminal and $\alpha_{\mathrm{inv}} = 0$, every adjusted score equals $r_{\mathrm{succ}}\gamma$. Each source group with $n_s > 1$ therefore has $\widehat{\mu}_{\mathrm{FE}}(s) = r_{\mathrm{succ}}\gamma$ and $\widehat{\sigma}_{\mathrm{FE}}(s) = 0$, giving $A^{\mathrm{FE}}_{m,t} = 0$. The singleton rule gives the same result for $n_s = 1$. If $\alpha_{\mathrm{inv}} > 0$, adjusted scores instead equal $r_{\mathrm{succ}}\gamma - \alpha_{\mathrm{inv}}\mathbf{1}[t \notin \mathcal{A}_m]$. Source groups containing both valid and invalid occurrences then have nonzero normalized credit.

Subsequent uses of A^{FE}, including the RMS scale in Appendix D.3, refer to this normalized component. The step penalty enters episode reward preprocessing before normalization. These score adjustments leave q_x, Z_β, and F_β unchanged. Disabling the invalid-action penalty sets $\alpha_{\mathrm{inv}} = 0$.

C.5 Formal verification in Lean

We formalize and verify Propositions 1–3, the shortest-distance limit, and related bounds in Lean 4 using mathlib. The Lean kernel accepts all formalized proofs. The proof code includes proof sources, theorem correspondence, and reproducible verification commands, and is available alongside our implementation in the anonymous repository.

D Structure-Aware Credit Refinement: Definitions and Properties

Mixed-group transfers preserve episode token mass within each positive trajectory. The definitions below specify the structural evidence and credit gates in Section 4.2. We fix a task x with M trajectories, where trajectory m contains states $s_{m,0:T_m}$ and transitions indexed by $0 \le t < T_m$. A positive trajectory satisfies $R(\tau_m) > 0$. We count positive trajectories by

$$H_x = \sum_{m=1}^{M} \mathbf{1}[R(\tau_m) > 0]. \tag{43}$$

The cases $H_x = 0$, $0 < H_x < M$, and $H_x = M$ denote all-failure, mixed-outcome, and all-positive groups, respectively.

D.1 Repeated closures and persistent exits

Structural matching uses the graph-state representations in Appendix F.2 and normalized executed actions. Action normalization strips surrounding whitespace, converts text to lowercase, and collapses consecutive whitespace. Let $\mathcal{A}_m$ contain positions that pass the environment-specific action-validity check. For WebShop, this check concerns action format and does not determine whether the environment can execute the action. Every observed transition updates the state prefix, including transitions outside $\mathcal{A}_m$.

Let $(v_0, \dots, v_K) = \mathrm{LE}(s_{m,0:t})$ be the chronological loop-erased prefix before transition t. A closure occurs when $s_{m,t+1} = v_j$ for some j. The loop footprint consists of the states $U_{m,t} = \{v_j, \dots, v_K\}$. Loop erasure retains the prefix $(v_0, \dots, v_j)$. The closing set $\mathcal{C}_m$ includes all such events; the repetition counts include only valid closures:

$$\begin{aligned} N^{\mathrm{cl}}_m(e; t) &= \sum_{v=0}^{t} \mathbf{1}\{v \in \mathcal{C}_m \cap \mathcal{A}_m,\ e_{m,v} = e\}, \\ M^{\mathrm{rep}}_x(e) &= \sum_{j=1}^{M} \mathbf{1}\{N^{\mathrm{cl}}_j(e; T_j - 1) \ge 2\}. \end{aligned} \tag{44}$$

Thus M_x^{rep} counts trajectories with repeated valid closures of the same transition, including the current trajectory. Define the eligible closing positions as

$$\mathcal{E}_m = \left\{ t \in \mathcal{C}_m \cap \mathcal{A}_m : N_m^{\mathrm{cl}}(e_{m,t}; t) \geq 2 \right\}. \tag{45}$$

The first valid closing occurrence has count one and lies outside this set. A repeated visit to a state alone does not increase this count unless the same valid closing transition occurs again.

A persistent exit leaves the footprint and stays outside it throughout the remaining observed suffix. For a closure at t, define

$$u_{m,t}^{\star} = \min \left\{ u : \begin{array}{ll} t < u < T_m, & u \in \mathcal{A}_m, \quad s_{m,u} \in U_{m,t}, \\ s_{m,v} \notin U_{m,t} & \text{for every } u+1 \leq v \leq T_m \end{array} \right\}. \tag{46}$$

The condition includes the immediate successor and the final observed state. If the set is empty, we leave $u_{m,t}^{\star}$ undefined and the closure supplies no exit witness. State continuity makes a qualifying exit the final departure from this footprint. The definition concerns the observed trajectory suffix.

In a mixed group, the structural witnesses are

$$\mathcal{J}_m = \begin{cases} \{t \in \mathcal{E}_m : u_{m,t}^{\star} \text{ exists}\}, & R(\tau_m) > 0, \\ \emptyset, & R(\tau_m) \leq 0. \end{cases} \tag{47}$$

Their persistent exits define the group-level transition set

$$\mathcal{R}_x = \left\{ e_{m,u_{m,t}^{\star}} : 1 \leq m \leq M,\ t \in \mathcal{J}_m \right\}. \tag{48}$$

This set depends on closure and exit evidence before any donor-sign test. A structurally supported exit can therefore enter $\mathcal{R}_x$ even if its paired closure has no positive episode mass to transfer.

D.2 Conservation of episode token mass

Transfers operate on the original episode credit of each step. Let $\chi_{m,t,k} \in \{0,1\}$ be the response mask and $A_{m,t,k}^{E,0}$ the episode advantage before structural refinement. Define the token count and original credit mass by

$$\ell_{m,t} = \sum_k \chi_{m,t,k} > 0, \qquad B_{m,t} = \sum_k \chi_{m,t,k} A_{m,t,k}^{E,0}. \tag{49}$$

The inherited updater broadcasts the same episode component over unmasked tokens of a step. At valid positions subject to refinement, this gives

$$B_{m,t} = \ell_{m,t} A^E(\tau_m). \tag{50}$$

The donor set selects structural witnesses with positive original mass:

$$\mathcal{D}_m^{+} = \{ t \in \mathcal{J}_m : B_{m,t} > 0 \}. \tag{51}$$

Eq. (8) removes each donor's original $B_{m,t}$ and adds it to position $u_{m,t}^{\star}$ in the same trajectory. We accumulate all removals and additions before updating any position. A position can donate its original mass while receiving mass from other donors, and multiple donors can share one recipient. Every transferred quantity uses the original B.

Proposition 3 (Conservation within positive trajectories)**.** *For every positive trajectory in a mixed group, simultaneous redistribution preserves its episode token mass:*

$$\sum_{t=0}^{T_m-1} B'_{m,t} = \sum_{t=0}^{T_m-1} B_{m,t}. \tag{52}$$

Proof. Fix a positive trajectory m in a mixed group. Every $t \in \mathcal{D}_m^{+}$ has one recipient $u_{m,t}^{\star} \in \{0, \ldots, T_m - 1\}$. Therefore

$$\sum_{u=0}^{T_m-1} \mathbf{1}[u_{m,t}^{\star} = u] = 1 \quad \text{for every } t \in \mathcal{D}_m^{+}. \tag{53}$$

Sum the update in Eq. (8) over all positions. All sums are finite, so we can exchange their order:

$$\begin{aligned}\sum_{u=0}^{T_m-1} B'_{m,u} &= \sum_{u=0}^{T_m-1} B_{m,u} - \sum_{t\in\mathcal{D}_m^+} B_{m,t} + \sum_{u=0}^{T_m-1}\sum_{t\in\mathcal{D}_m^+} \mathbf{1}[u^\star_{m,t}=u]B_{m,t} \\ &= \sum_{u=0}^{T_m-1} B_{m,u} - \sum_{t\in\mathcal{D}_m^+} B_{m,t} + \sum_{t\in\mathcal{D}_m^+} B_{m,t} \underbrace{\sum_{u=0}^{T_m-1} \mathbf{1}[u^\star_{m,t}=u]}_{=1} \\ &= \sum_{u=0}^{T_m-1} B_{m,u}.\end{aligned} \tag{54}$$

The cancellation requires one recipient per donor, but it does not require different donors to have different recipients. If a recipient also donates, its removal uses its own original mass and its incoming additions use the other donors' original masses. Thus the same calculation covers both multiple donors sharing a recipient and positions that both receive and donate.

We convert the updated mass back to a per-token advantage by

$$\widetilde{A}^E_{m,t} = \frac{B'_{m,t}}{\ell_{m,t}}, \qquad \sum_k \chi_{m,t,k}\widetilde{A}^E_{m,t} = \ell_{m,t}\frac{B'_{m,t}}{\ell_{m,t}} = B'_{m,t}. \tag{55}$$

The positive token count makes this division well-defined. Summing the second identity over steps proves that broadcasting the refined advantages preserves the same total episode token mass. □

For trajectories with non-positive outcomes, the mixed branch separately clears conflicting negative episode credit. It matches the full transition key against $\mathcal{R}_x$ and applies

$$B'_{m,t} = \begin{cases}\max\{B_{m,t}, 0\}, & t\in\mathcal{A}_m,\ e_{m,t}\in\mathcal{R}_x, \\ B_{m,t}, & \text{otherwise},\end{cases} \tag{56}$$

where this equation covers only non-positive trajectories. The matching rule uses transition identity independently of closure status. At a matching valid position, the update removes the negative part of the original episode mass:

$$B'_{m,t} - B_{m,t} = \max\{-B_{m,t}, 0\}. \tag{57}$$

It leaves a nonnegative mass unchanged. Positions outside the matching set retain their original mass. This operation changes only the episode component and leaves free-energy credit intact.

D.3 All-failure penalties and inactive cases

The all-failure branch scales repeated-closing penalties using valid step-level free-energy advantages. Define $\mathcal{I}_x = \{(m,t) : t\in\mathcal{A}_m,\ |A^{\mathrm{FE}}_{m,t}| > \varepsilon\}$. Each occurrence contributes once, with either sign, and the implementation uses $\varepsilon = 10^{-6}$ by default:

$$\rho_x = \begin{cases}\sqrt{\dfrac{1}{|\mathcal{I}_x|}\displaystyle\sum_{(m,t)\in\mathcal{I}_x}(A^{\mathrm{FE}}_{m,t})^2}, & |\mathcal{I}_x| > 0, \\ 1, & |\mathcal{I}_x| = 0.\end{cases} \tag{58}$$

The shortest-path variant preserves source-state grouping, invalid-action handling, and advantage normalization. For the shortest-path variant, the same RMS rule uses its normalized, invalid-action-adjusted shortest-path graph advantages, with the same eligibility criteria and fallback value. For $t\in\mathcal{E}_m$, define $\delta_{m,t} = N^{\mathrm{cl}}_m(e_{m,t};t) + M^{\mathrm{rep}}_x(e_{m,t}) - 2$ and $w_{m,t} = \delta_{m,t}/(\delta_{m,t}+1)$. Eligibility implies that the current trajectory contributes to $M^{\mathrm{rep}}_x(e_{m,t})$. Hence

$$N^{\mathrm{cl}}_m(e_{m,t};t) \geq 2, \qquad M^{\mathrm{rep}}_x(e_{m,t}) \geq 1, \qquad \delta_{m,t}\geq 1, \qquad \frac{1}{2} \leq w_{m,t} < 1. \tag{59}$$

As a result, the denominator is positive and the repetition weight remains bounded. Thus, the complete domain of Eq. (7) is

$$\Delta^{\text{fail}}_{m,t} = \begin{cases} -\rho_x w_{m,t}, & H_x = 0,\ t \in \mathcal{E}_m, \\ 0, & \text{otherwise.} \end{cases} \tag{60}$$

Outside mixed groups, the episode component retains its original value:

$$\widetilde{A}^{E}_{m,t} = \frac{B_{m,t}}{\ell_{m,t}}. \tag{61}$$

An all-failure group without eligible closures receives zero correction. A mixed group without exit witnesses leaves both components unchanged. Within a mixed group, an empty donor set causes no transfer, and the absence of valid matching negative episode credit causes no clipping. All-positive groups preserve the original advantages. All branches preserve the free-energy component itself.

Unified structure-aware advantage. The refined episode credit and the local correction form one structure-aware component:

$$A^{\text{SC}}_{m,t} = \widetilde{A}^{E}_{m,t} + \Delta^{\text{fail}}_{m,t}. \tag{62}$$

The scale ρ_x comes from the free-energy advantages. Section 4.3 combines this structure-aware component with free-energy credit using two outer weighting coefficients. For positive trajectories in mixed groups, $\Delta^{\text{fail}}_{m,t} = 0$, and Proposition 3 gives $\sum_t \ell_{m,t} A^{\text{SC}}_{m,t} = \sum_t B_{m,t}$.

Ablation without structure-aware refinement. This ablation disables both mixed-outcome refinement and all-failure correction while retaining the original episode advantage, including the inherited reward preprocessing and normalization. We set $\widetilde{A}^{E} = A^{E}$ and $\Delta^{\text{fail}} = 0$, with both outer weights remaining one, yielding

$$A_{\text{w/o SC}} = A^{\text{FE}} + A^{E}.$$

E Policy Optimization

Section 4.3 presents the core clipped objective with a target KL penalty. This appendix specifies the token-level loss used by the inherited updater, including negative-advantage dual clipping, entropy regularization, and the sampled reference-policy penalty.

E.1 Token advantages, masks, and policy ratios

Within an optimization batch, m uniquely indexes trajectory occurrences across tasks; structural matching still takes place separately within each task. At response token k of step (m, t), let $h_{m,t,k}$ contain the policy input and preceding response tokens, and let $y_{m,t,k}$ denote the sampled token. The response mask $\chi_{m,t,k} \in \{0, 1\}$ selects tokens that contribute to the loss. This mask can include tokens from responses whose executed actions fail the validity check in Appendix D.1. For every unmasked token, the step-level advantage gives

$$A^{\text{tok}}_{m,t,k} = A^{\text{FESGPO}}_{m,t} = \lambda_{\text{FE}} A^{\text{FE}}_{m,t} + \lambda_{\text{SC}} A^{\text{SC}}_{m,t}. \tag{63}$$

Equation (62) includes the all-failure correction within $A^{\text{SC}}_{m,t}$. The coefficients $\lambda_{\text{FE}}, \lambda_{\text{SC}} \geq 0$ follow Section 4.3; both equal one in all reported experiments. The token count in Section 4.2 is $\ell_{m,t} = \sum_k \chi_{m,t,k}$; the response mask and the action-validity gate serve different roles.

During each update, we hold these advantages and the stored old-policy token log probabilities fixed. The current policy determines the ratio

$$\begin{aligned} r_{m,t,k}(\theta) &= \frac{\pi_\theta(y_{m,t,k} \mid h_{m,t,k})}{\pi_{\text{old}}(y_{m,t,k} \mid h_{m,t,k})} \\ &= \exp(\log \pi_\theta(y_{m,t,k} \mid h_{m,t,k}) - \log \pi_{\text{old}}(y_{m,t,k} \mid h_{m,t,k}))\,. \end{aligned} \tag{64}$$

Each ratio compares probabilities of the same sampled token under the same context. The stored denominator remains fixed while the numerator changes during optimization.

E.2 Clipped surrogate and negative-advantage dual clipping

For brevity, write $A = A^{\mathrm{FESGPO}}_{m,t}$ and $r = r_{m,t,k}(\theta)$. Let $\epsilon^{-}_{\mathrm{clip}}, \epsilon^{+}_{\mathrm{clip}} > 0$ be the lower and upper clipping widths. Define

$$\bar{r} = \mathrm{clip}(r, 1-\epsilon^{-}_{\mathrm{clip}}, 1+\epsilon^{+}_{\mathrm{clip}}), \quad g(r, A) \quad = \min\{rA, \bar{r}A\}. \tag{65}$$

The vanilla updater applies an additional bound for negative advantages, with $c_{\mathrm{dual}} > 1$:

$$\widetilde{g}(r, A) = \begin{cases} \max\{g(r, A), c_{\mathrm{dual}}A\}, & A < 0, \\ g(r, A), & A \geq 0. \end{cases} \tag{66}$$

The updater minimizes $-\widetilde{g}$. For $A < 0$, this equals $\min\{\max(-rA, -\bar{r}A), -c_{\mathrm{dual}}A\}$ and caps the surrogate loss at $-c_{\mathrm{dual}}A$. Nonnegative advantages retain the original clipped surrogate in Eq. (9).

E.3 Loss aggregation and regularization

For a microbatch $\mathcal{B}$, define its number of contributing tokens and the masked token mean by

$$\begin{aligned} N_{\mathcal{B}} &= \sum_{(m,t,k)\in\mathcal{B}} \chi_{m,t,k} > 0, \\ \langle f \rangle_{\mathcal{B},\chi} &= \frac{1}{N_{\mathcal{B}}} \sum_{(m,t,k)\in\mathcal{B}} \chi_{m,t,k} f_{m,t,k}. \end{aligned} \tag{67}$$

Each batch entry contains the prompt and generated response for one environment step; mini-batch size counts these entries. Each microbatch uses its own masked token mean. For fixed microbatches, the updater divides the loss by the configured gradient-accumulation count. For dynamic batching, it multiplies the loss by the number of entries in the microbatch divided by the configured mini-batch size. The resulting loss need not equal a single masked token mean over the full mini-batch. The expectation over $\mathcal{B}$ in Eq. (10) uses the same microbatch weighting.

The full actor loss is

$$\mathcal{L}_{\mathcal{B}}(\theta) = -\langle \widetilde{g}(r, A) \rangle_{\mathcal{B},\chi} - \alpha_{\mathrm{ent}} \langle \mathcal{H}_\theta \rangle_{\mathcal{B},\chi} + \alpha_{\mathrm{KL}} \langle \widehat{K}_\theta \rangle_{\mathcal{B},\chi}, \tag{68}$$

where $\alpha_{\mathrm{ent}}, \alpha_{\mathrm{KL}} \geq 0$. A disabled regularizer has coefficient zero. With token vocabulary $\mathcal{V}$, the conditional entropy is

$$\mathcal{H}_\theta(h) = -\sum_{v\in\mathcal{V}} \pi_\theta(v \mid h) \log \pi_\theta(v \mid h). \tag{69}$$

Target KL and sampled penalty. The KL shorthand in Eq. (10) denotes the conditional divergence averaged over rollout contexts with the same mask and microbatch weights:

$$D_{\mathrm{KL}}(\pi_\theta \| \pi_{\mathrm{ref}}) := \mathbb{E}_{\mathcal{B}} \left[\frac{1}{N_{\mathcal{B}}} \sum_{(m,t,k)\in\mathcal{B}} \chi_{m,t,k} \sum_{v\in\mathcal{V}} \pi_\theta(v \mid h_{m,t,k}) \log \frac{\pi_\theta(v \mid h_{m,t,k})}{\pi_{\mathrm{ref}}(v \mid h_{m,t,k})} \right]. \tag{70}$$

This defines the target regularizer, not a vocabulary sum evaluated by the updater. For the implemented low-variance sampled penalty, write

$$\begin{aligned} \zeta_{m,t,k}(\theta) &= \log \pi_{\mathrm{ref}}(y_{m,t,k} \mid h_{m,t,k}) - \log \pi_\theta(y_{m,t,k} \mid h_{m,t,k}), \\ \widehat{K}_{m,t,k}(\theta) &= \mathrm{clip}\Big(e^{\zeta_{m,t,k}(\theta)} - \zeta_{m,t,k}(\theta) - 1, -10, 10 \Big). \end{aligned} \tag{71}$$

The updater evaluates this surrogate on stored rollout tokens and applies the displayed numerical clipping. Without clipping, its expectation equals the conditional KL when tokens follow the current policy. Stored old-policy samples and clipping generally change that expectation, so the implemented penalty is a sampled surrogate for the target KL, not an exact unbiased estimate at arbitrary θ. The old policy supplies the importance-ratio denominator; the reference policy supplies the regularization target. Both remain fixed during the update.

The free-energy inverse temperature $\beta = -\log\gamma$, the KL coefficient α_{KL}, the normalization stabilizer ε, and the PPO clipping widths $\epsilon^{\pm}_{\mathrm{clip}}$ have distinct roles. FESGPO changes the supplied advantages and retains the inherited policy updater.

F Implementation Details

F.1 Computing Details

All FESGPO training experiments use a fixed budget of 150 training iterations across ALFWorld and WebShop for both model sizes. For experiments using Qwen2.5-1.5B-Instruct, we perform training on two NVIDIA H200 GPUs. For experiments using the larger Qwen2.5-7B-Instruct model, we employ four NVIDIA H200 GPUs.

F.2 Prompts and graph states

We use the environment-specific prompt templates in Figures 7 and 8. Each prompt supplies the task information, the current observation, and the admissible actions. When history is available, we include up to two recent observation-action pairs. We ask the policy to enclose its reasoning in `<think>...</think>` tags and one chosen action in `<action>...</action>` tags. For ALFWorld, the current observation also includes the agent's location, held-item status, and object-movement records when available. The figures show the history-conditioned templates; the initial step uses the corresponding no-history variant.

Graph-state representation. We construct graph states from the following environment information.

Environment	Graph-state representation
ALFWorld	Current textual observation, agent location, held-item identity and processing status, object-movement/status records, and sorted admissible commands.
WebShop	Formatted current page observation, including page controls and selections represented in the observation text.

Within each task-specific rollout group, we merge identical serialized state representations. Both credit components use these representations after the success-terminal mapping in Section 4.1. In Section 3.1, policy conditioning is shorthand for the full prompt associated with each transition occurrence (Figures 7 and 8); Appendix E.1 specifies the token-level context. Graph construction and structural matching use the representations listed in the table. ALFWorld's object records retain information from previous interactions. For structural matching, we pair each state representation with its rollout-group identifier and retain step indices as temporal metadata.

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}].

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 7: 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 of the current situation are: [{available_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 8: Prompt template used for WebShop experiments.

F.3 Comparing Methods

We compare FESGPO with prompting-based agents and reinforcement learning methods spanning trajectory-level, state-grouped, history-grouped, graph-based, and milestone-based approaches.

Prompting baselines. These baselines include GPT-4o, GPT-5.6-sol, Gemini-3.5-Flash, Kimi-K3, and GLM-5.2, as well as Qwen2.5 models without task-specific reinforcement learning.

ReAct. This prompting-based framework interleaves reasoning and environment actions during multi-turn task execution (Yao et al., 2023).

Reflexion. This framework uses natural-language reflections on previous attempts to guide subsequent decisions, without updating model parameters (Shinn et al., 2023).

PPO. Proximal Policy Optimization is an actor-critic algorithm that estimates advantages with a value function and constrains policy updates through a clipped objective (Schulman et al., 2017).

RLOO. REINFORCE Leave-One-Out estimates each trajectory's advantage against the mean return of the other trajectories sampled for the same task, providing a low-variance baseline without learning a value network (Ahmadian et al., 2024).

GRPO. Group Relative Policy Optimization computes trajectory-level advantages from within-group returns and updates the policy without a separate value network (Shao et al., 2024).

GiGPO. Group-in-Group Policy Optimization combines episode-level and step-level advantages. Its anchor state grouping mechanism compares actions that originate from the same environment state across trajectories (Feng et al., 2025).

GraphGPO. GraphGPO organizes sampled trajectories into a state-transition graph and derives graph credit from shortest paths to successful terminal states. It combines this signal with episode-level advantages (Cheng et al., 2026).

HGPO. Hierarchy-of-Groups Policy Optimization organizes steps into hierarchical groups according to historical-context consistency and adaptively aggregates advantages across multiple hierarchy levels (He et al., 2026a).

BEACON. BEACON partitions trajectories at milestone boundaries, shapes rewards within segments, and combines advantages at two scales to credit local progress (Wang et al., 2026b).

GraphGPO with Dynamic Sampling. We follow GraphGPO's dynamic-sampling setup (Cheng et al., 2026) and retain its credit computation. Within each optimization step, each sampling round generates 16 task groups with eight trajectories per group. We discard groups with identical episode

returns and accumulate the remaining trajectories. Sampling stops when the retained pool contains at least 128 trajectories or after ten rounds, including the initial round. If sampling reaches the tenth round, we retain all groups from that round. FESGPO uses one unfiltered round of 128 trajectories per update. Cumulative rollout counts include all trajectories generated during training, including those discarded by dynamic sampling.

We use publicly available implementations for the reinforcement learning baselines that we reproduce. We adopt the PPO and RLOO results reported by GraphGPO (Cheng et al., 2026).

F.4 ALFWORLD

All models that we train for the main results share the following base training configuration and retain their method-specific hyperparameters. We set the maximum prompt and response lengths to 2048 and 512 tokens, respectively, and allow up to 50 environment steps per episode. The actor learning rate is 1×10^{-6}. Successful task completion yields a task reward of 10, and failure yields 0. We apply a separate reward penalty of -0.1 to invalid actions.

For group-based reinforcement learning methods, each rollout batch contains 16 task groups with 8 trajectories per group. We set the training and test sampling temperatures to 1.0 and 0.4, respectively. The PPO mini-batch size is 256, and the KL-divergence loss coefficient is 0.01.

For GiGPO, we set the discount factor λ to 0.95. For GraphGPO, we set the distance discount factor ω to 0.10. HGPO uses a history length of 2 and length-based weighting with an exponent of 1. For BEACON, both the discount factor and the milestone-matching threshold are 0.95. For FESGPO, we set the distance discount factor γ to 0.20 and assign a weight of 1 to both free-energy credit and structure-aware credit.

All models we train use the same augmented ALFWorld environment as GraphGPO (Cheng et al., 2026). This environment adds the agent's current location and held-object information to the original text observations. At each decision, the policy receives up to two recent observation-action pairs.

F.5 WEBSHOP

We use a common base configuration for all models that we train for the main results, while retaining method-specific hyperparameters. The maximum prompt and response lengths are 5120 and 512 tokens, respectively, and each episode allows up to 15 environment steps. The actor learning rate is 1×10^{-6}. We assign a task reward of 10 for successful task completion and 0 otherwise, with a separate reward penalty of -0.1 for invalid actions.

As with ALFWorld, group-based reinforcement learning methods sample 16 task groups per rollout batch, with 8 trajectories per group, totaling 128 trajectories. The training and test sampling temperatures are 1.0 and 0.4, respectively. The mini-batch size is 64, and the KL-divergence loss coefficient is 0.01. For GiGPO, we set the discount factor λ to 0.95. HGPO uses a history length of 2 and length-based weighting with an exponent of 1. For BEACON, we set the discount factor to 0.95.

We use the official *text-rich* WebShop environment, following GraphGPO (Cheng et al., 2026). This version provides additional information about interactive elements, such as buttons. All models that we train use this environment with up to two recent observation-action pairs. The distance discount factor ω in GraphGPO is 0.20. For FESGPO, we set $\gamma = 0.10$ for Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct, with a weight of 1 for both free-energy credit and structure-aware credit.

G CASE STUDY

We present two successful evaluation trajectories of FESGPO on ALFWorld and WebShop, using Qwen2.5-1.5B-Instruct at training step 150. We select these episodes for their complete records and concise task execution. The boxes show every task-facing prompt and the original model response; we omit the shared system message and chat role labels. Gold text marks the current observation and admissible actions, while green and magenta mark reasoning and action tags. These examples illustrate individual successful executions.

G.1 ALFWORLD

The agent takes a dishsponge from a toilet, cleans it at a sinkbasin, and places it in a cart. The episode reaches strict success in six actions.

Prompt (ALFWorld, Step 1)

You are an expert agent operating in the ALFRED Embodied Environment. Your current observation is: -= Welcome to TextWorld, ALFRED! =- You are in the middle of a room. Looking quickly around you, you see a bathtubbasin 1, a cart 1, a garbagecan 1, a handtowelholder 2, a handtowelholder 1, a shelf 4, a shelf 3, a shelf 2, a shelf 1, a sinkbasin 1, a toilet 1, a toiletpaperhanger 1, a towelholder 2, and a towelholder 1. Your task is to: put a clean dishsponge in cart. Location: middle of a room. Items in hand (status): nothing. Your admissible actions of the current situation are: ['go to bathtubbasin 1' 'go to cart 1' 'go to garbagecan 1' 'go to handtowelholder 1' 'go to handtowelholder 2' 'go to shelf 1' 'go to shelf 2' 'go to shelf 3' 'go to shelf 4' 'go to sinkbasin 1' 'go to toilet 1' 'go to toiletpaperhanger 1' 'go to towelholder 1' 'go to towelholder 2' '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> The task is to put a clean dishsponge in the cart. I need to find a dishsponge, clean it, and then put it in the cart. </think>
<action>go to toilet 1</action>

Prompt (ALFWorld, Step 2)

You are an expert agent operating in the ALFRED Embodied Environment. Your task is to: put a clean dishsponge in cart. 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! =- You are in the middle of a room. Looking quickly around you, you see a bathtubbasin 1, a cart 1, a garbagecan 1, a handtowelholder 2, a handtowelholder 1, a shelf 4, a shelf 3, a shelf 2, a shelf 1, a sinkbasin 1, a toilet 1, a toiletpaperhanger 1, a towelholder 2, and a towelholder 1. Your task is to: put a clean dishsponge in cart. Location: middle of a room. Items in hand (status): nothing. ', Action 1: 'go to toilet 1'] You are now at step 2 and your current observation is: You arrive at toilet 1. On the toilet 1, you see a candle 1, a dishsponge 1, a papertowelroll 1, a spraybottle 2, and a spraybottle 1. Location: toilet 1. Items in hand (status): nothing. Your admissible actions of the current situation are: ['examine toilet 1' 'go to bathtubbasin 1' 'go to cart 1' 'go to garbagecan 1' 'go to handtowelholder 1' 'go to handtowelholder 2' 'go to shelf 1' 'go to shelf 2' 'go to shelf 3' 'go to shelf 4' 'go to sinkbasin 1' 'go to toiletpaperhanger 1' 'go to towelholder 1' 'go to towelholder 2' 'inventory' 'look' 'take candle 1 from toilet 1' 'take dishsponge 1 from toilet 1' 'take papertowelroll 1 from toilet 1' 'take spraybottle 1 from toilet 1' 'take spraybottle 2 from toilet 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> The current situation is that you are at toilet 1. You have not yet taken any action to pick up the dishsponge 1. You need to go to the place where the dishsponge is located and then take it. </think>
<action>take dishsponge 1 from toilet 1</action>

Prompt (ALFWorld, Step 3)

You are an expert agent operating in the ALFRED Embodied Environment. Your task is to: put a clean dishsponge in cart. 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! =- You are in the middle of a room. Looking quickly around you, you see a bathtubbasin 1, a cart 1, a garbagecan 1, a handtowelholder 2, a handtowelholder 1, a shelf 4, a shelf 3, a shelf 2, a shelf 1, a sinkbasin 1, a toilet 1, a toiletpaperhanger 1, a towelholder 2, and a towelholder 1. Your task is to: put a clean dishsponge in cart. Location: middle of a room. Items in hand (status): nothing. ', Action 1: 'go to toilet 1'] [Observation 2: 'You arrive at toilet 1. On the toilet 1, you see a candle 1, a dishsponge 1, a papertowelroll 1, a spraybottle 2, and a spraybottle 1. Location: toilet 1. Items in hand (status): nothing. ', Action 2: 'take dishsponge 1 from toilet 1'] You are now at step 3 and your current observation is: You pick up the dishsponge 1 from the toilet 1. Location: toilet 1. Items in hand (status): dishsponge 1(unprocessed). Your admissible actions of the current situation are: ['examine dishsponge 1' 'examine toilet 1' 'go to bathtubbasin 1' 'go to cart 1' 'go to garbagecan 1' 'go to handtowelholder 1' 'go to handtowelholder 2' 'go to shelf 1' 'go to shelf 2' 'go to shelf 3' 'go to shelf 4' 'go to sinkbasin 1' 'go to toiletpaperhanger 1' 'go to towelholder 1' 'go to towelholder 2' 'inventory' 'look' 'move dishsponge 1 to toilet 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> The current situation is that I have just picked up the dishsponge 1 from the toilet 1. My task is to put a clean dishsponge in the cart. I need to ensure that the dishsponge is clean before placing it in the cart. </think>
<action>go to sinkbasin 1</action>

Prompt (ALFWorld, Step 4)

You are an expert agent operating in the ALFRED Embodied Environment. Your task is to: put a clean dishsponge in cart. 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 toilet 1. On the toilet 1, you see a candle 1, a dishsponge 1, a papertowelroll 1, a spraybottle 2, and a spraybottle 1. Location: toilet 1. Items in hand (status): nothing. ', Action 2: 'take dishsponge 1 from toilet 1'] [Observation 3: 'You pick up the dishsponge 1 from the toilet 1. Location: toilet 1. Items in hand (status): dishsponge 1(unprocessed). ', Action 3: 'go to sinkbasin 1'] You are now at step 4 and your current observation is: You arrive at sinkbasin 1. On the sinkbasin 1, you see nothing. Location: sinkbasin 1. Items in hand (status): dishsponge 1(unprocessed). Your admissible actions of the current situation are: ['clean dishsponge 1 with sinkbasin 1' 'examine dishsponge 1' 'examine sinkbasin 1' 'go to bathtubbasin 1' 'go to cart 1' 'go to garbagecan 1' 'go to handtowelholder 1' 'go to handtowelholder 2' 'go to shelf 1' 'go to shelf 2' 'go to shelf 3' 'go to shelf 4' 'go to toilet 1' 'go to toiletpaperhanger 1' 'go to towelholder 1' 'go to towelholder 2' 'inventory' 'look' 'move dishsponge 1 to sinkbasin 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> The current situation is that you are at the sinkbasin 1, and there is a dishsponge 1 in your hand. You need to clean the dishsponge 1. </think>
<action>clean dishsponge 1 with sinkbasin 1</action>

Prompt (ALFWorld, Step 5)

You are an expert agent operating in the ALFRED Embodied Environment. Your task is to: put a clean dishsponge in cart. 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 dishsponge 1 from the toilet 1. Location: toilet 1. Items in hand (status): dishsponge 1(unprocessed). ', Action 3: 'go to sinkbasin 1'] [Observation 4: 'You arrive at sinkbasin 1. On the sinkbasin 1, you see nothing. Location: sinkbasin 1. Items in hand (status): dishsponge 1(unprocessed). ', Action 4: 'clean dishsponge 1 with sinkbasin 1'] You are now at step 5 and your current observation is: You clean the dishsponge 1 using the sinkbasin 1. Location: sinkbasin 1. Items in hand (status): dishsponge 1(cleaned). Your admissible actions of the current situation are: ['clean dishsponge 1 with sinkbasin 1' 'examine dishsponge 1' 'examine sinkbasin 1' 'go to bathtubbasin 1' 'go to cart 1' 'go to garbagecan 1' 'go to handtowelholder 1' 'go to handtowelholder 2' 'go to shelf 1' 'go to shelf 2' 'go to shelf 3' 'go to shelf 4' 'go to toilet 1' 'go to toiletpaperhanger 1' 'go to towelholder 1' 'go to towelholder 2' 'inventory' 'look' 'move dishsponge 1 to sinkbasin 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> The current situation is that you have cleaned the dishsponge 1 using the sinkbasin 1. You need to put the clean dishsponge 1 in the cart 1. </think>
<action>go to cart 1</action>

Prompt (ALFWorld, Step 6)

You are an expert agent operating in the ALFRED Embodied Environment. Your task is to: put a clean dishsponge in cart. 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 sinkbasin 1. On the sinkbasin 1, you see nothing. Location: sinkbasin 1. Items in hand (status): dishsponge 1(unprocessed). ', Action 4: 'clean dishsponge 1 with sinkbasin 1'] [Observation 5: 'You clean the dishsponge 1 using the sinkbasin 1. Location: sinkbasin 1. Items in hand (status): dishsponge 1(cleaned). ', Action 5: 'go to cart 1'] You are now at step 6 and your current observation is: You arrive at cart 1. On the cart 1, you see a cloth 2, a cloth 1, a soapbottle 1, and a spraybottle 3. Location: cart 1. Items in hand (status): dishsponge 1(cleaned). Your admissible actions of the current situation are: ['examine cart 1' 'examine dishsponge 1' 'go to bathtubbasin 1' 'go to garbagecan 1' 'go to handtowelholder 1' 'go to handtowelholder 2' 'go to shelf 1' 'go to shelf 2' 'go to shelf 3' 'go to shelf 4' 'go to sinkbasin 1' 'go to toilet 1' 'go to toiletpaperhanger 1' 'go to towelholder 1' 'go to towelholder 2' 'inventory' 'look' 'move dishsponge 1 to cart 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> The current situation is that you have cleaned the dishsponge 1 and are at cart 1. You need to put the cleaned dishsponge 1 in the cart. </think>
<action>move dishsponge 1 to cart 1</action>

Outcome: Strict success.

G.2 WEBSHOP

The agent searches for sleepwear, opens a product page, selects the requested color and size, and completes the purchase. The episode reaches strict success in five actions, with a Task Score of 100.

Prompt (WebShop, Step 1)

You are an expert autonomous agent operating in the WebShop e-commerce environment. Your task is to: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars. Your current observation is: WebShop Instruction: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Search [button_] . 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> To find the men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars, I need to search for products that meet these criteria. I will look for products with the given color and size, and compare their prices to find the lowest-priced option. </think> <action>search[multi 7 long sleeve men's sleep & lounge, size: large]</action>

Prompt (WebShop, Step 2)

You are an expert autonomous agent operating in the WebShop e-commerce environment. Your task is to: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 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 Instruction: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Search [button_] ', Action 1: 'search[multi 7 long sleeve men's sleep & lounge, size: large]'] You are now at step 2 and your current observation is: Instruction: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Back to Search [button_] Page 1 (Total results: 50) [button] Next > [button_] [button] B09ND8P2QR [button_] InterestPrint Deer and Stars Men's Loungewear Pajama Sets - Long Sleeve Tee and Jogger Pant $43.59 [button] B09ND9DP7J [button_] InterestPrint Gold Horse Pattern Men's 2-Piece Sleepwear Set, Long Sleeve Shirt with Pants Loungewear $43.59 [button] B09NDS8F4V [button_] AODONG Onesie Pajamas for Women Onesies Romper Pajamas Printed Bodycon Jumpsuit Shorts Sexy One Piece Pjs Overall $2.99 to $7.99 [button] B09QQP3356 [button_] HAUKLIE Men's Sports Waffle Ribbed Polo Shirts Summer Short Sleeve Cotton Muscle Quarter-Zip Henley T-Shirt Tunics Tops $10.99 [button] B09NSC5VDG [button_] CofeeMO Womens Lingerie Sleep Lounge Everyday Nursing Bras Printed Underwear Seamless Printed Bralette for Sex Naughty Play $14.99 [button] B09S3BN15C [button_] Mens Linen Shirt,Men's Striped Shirts Casual Short Sleeve Button Down Shirts Regular Fit Hawaiian Shirts Beach Tees Tops $3.78 to $11.38 [button] B09R9YCM6R [button_] XXBR Summer T-shirts for Mens, Soldier Short Sleeve 3D Street Vintage Printed Shirt Slim Fit Muscle Casual Tee Tops $8.98 to $11.99 [button] B09KP78G37 [button_] Women Faux Fur Lined Jacket Coat Winter Warm Thick Fleece Outwear Trench Zipper Plus Size Long Sleeve Plush Overcoat $47.41 to $59.07 [button] B00O30JLDK [button_] Legendary Whitetails Men's Non-Typical Long Sleeve T-Shirt $10.52 to $40.5 [button] B09Q67H373 [button_] JSPOYOU 2022 Newly T-Shirt for Mens Funny 3D Graphics Pattern Crewneck Short Sleeve Tees Big and Tall Summer Casual Comfy Top $1.99 to $8.99 . Your admissible actions of the current situation are: ['click[back to search]', 'click[next >]', 'click[b09nd8p2qr]', 'click[b09nd9dp7j]', 'click[b09nds8f4v]', 'click[b09qqp3356]', 'click[b09nsc5vdg]', 'click[b09s3bn15c]', 'click[b09r9ycm6r]', 'click[b09kp78g37]', 'click[b00o30jldk]', 'click[b09q67h373]',]. 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> To find men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars, I need to search for products that match these criteria. I will check the search results for products with these specific features and prices. </think> <action>click[b09nd8p2qr]</action>

Prompt (WebShop, Step 3)

You are an expert autonomous agent operating in the WebShop e-commerce environment. Your task is to: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 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 Instruction: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Search [button_] ', Action 1: 'search[multi 7 long sleeve men's sleep & lounge, size: large]'] [Observation 2: 'Instruction: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Back to Search [button_] Page 1 (Total results: 50) [button] Next > [button_] [button] B09ND8P2QR [button_] InterestPrint Deer and Stars Men's Loungewear Pajama Sets - Long Sleeve Tee and Jogger Pant $43.59 [button] B09ND9DP7J [button_] InterestPrint Gold Horse Pattern Men's 2-Piece Sleepwear Set, Long Sleeve Shirt with Pants Loungewear $43.59 [button] B09NDS8F4V [button_] AODONG Onesie Pajamas for Women Onesies Romper Pajamas Printed Bodycon Jumpsuit Shorts Sexy One Piece Pjs Overall $2.99 to $7.99 [button] B09QQP3356 [button_] HAUKLIE Men's Sports Waffle Ribbed Polo Shirts Summer Short Sleeve Cotton Muscle Quarter-Zip Henley T-Shirt Tunics Tops $10.99 [button] B09NSC5VDG [button_] CofeeMO Womens Lingerie Sleep Lounge Everyday Nursing Bras Printed Underwear Seamless Printed Bralette for Sex Naughty Play $14.99 [button] B09S3BN15C [button_] Mens Linen Shirt,Men's Striped Shirts Casual Short Sleeve Button Down Shirts Regular Fit Hawaiian Shirts Beach Tees Tops $3.78 to $11.38 [button] B09R9YCM6R [button_] XXBR Summer T-shirts for Mens, Soldier Short Sleeve 3D Street Vintage Printed Shirt Slim Fit Muscle Casual Tee Tops $8.98 to $11.99 [button] B09KP78G37 [button_] Women Faux Fur Lined Jacket Coat Winter Warm Thick Fleece Outwear Trench Zipper Plus Size Long Sleeve Plush Overcoat $47.41 to $59.07 [button] B00O30JLDK [button_] Legendary Whitetails Men's Non-Typical Long Sleeve T-Shirt $10.52 to $40.5 [button] B09Q67H373 [button_] JSPOYOU 2022 Newly T-Shirt for Mens Funny 3D Graphics Pattern Crewneck Short Sleeve Tees Big and Tall Summer Casual Comfy Top $1.99 to $8.99 ', Action 2: 'click[b09nd8p2qr]'] You are now at step 3 and your current observation is: Instruction: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Back to Search [button_] [button] < Prev [button_] size [button] small [button_] [button] medium [button_] [button] large [button_] [button] x-large [button_] [button] xx-large [button_] color [button] multi 1 [button_] [button] multi 10 [button_] [button] multi 2 [button_] [button] multi 3 [button_] [button] multi 4 [button_] [button] multi 5 [button_] [button] multi 6 [button_] [button] multi 7 [button_] [button] multi 8 [button_] [button] multi 9 [button_] InterestPrint Deer and Stars Men's Loungewear Pajama Sets - Long Sleeve Tee and Jogger Pant Price: $43.59 Rating: N.A. [button] Description [button_] [button] Features [button_] [button] Reviews [button_] [button] Buy Now [button_] . Your admissible actions of the current situation are: ['click[back to search]', 'click[< prev]', 'click[description]', 'click[features]', 'click[reviews]', 'click[buy now]', 'click[small]', 'click[medium]', 'click[large]', 'click[x-large]', 'click[xx-large]', 'click[multi 1]', 'click[multi 10]', 'click[multi 2]', 'click[multi 3]', 'click[multi 4]', 'click[multi 5]', 'click[multi 6]', 'click[multi 7]', 'click[multi 8]', 'click[multi 9]',]. 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> To find men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars, I need to search for products that meet these criteria. Let's continue my search by filtering the results by color and size. </think> <action>click[multi 7]</action>

Prompt (WebShop, Step 4)

You are an expert autonomous agent operating in the WebShop e-commerce environment. Your task is to: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 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: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Back to Search [button_] Page 1 (Total results: 50) [button] Next > [button_] [button] B09ND8P2QR [button_] InterestPrint Deer and Stars Men's Loungewear Pajama Sets - Long Sleeve Tee and Jogger Pant $43.59 [button] B09ND9DP7J [button_] InterestPrint Gold Horse Pattern Men's 2-Piece Sleepwear Set, Long Sleeve Shirt with Pants Loungewear $43.59 [button] B09NDS8F4V [button_] AODONG Onesie Pajamas for Women Onesies Romper Pajamas Printed Bodycon Jumpsuit Shorts Sexy One Piece Pjs Overall $2.99 to $7.99 [button] B09QQP3356 [button_] HAUKLIE Men's Sports Waffle Ribbed Polo Shirts Summer Short Sleeve Cotton Muscle Quarter-Zip Henley T-Shirt Tunics Tops $10.99 [button] B09NSC5VDG [button_] CofeeMO Womens Lingerie Sleep Lounge Everyday Nursing Bras Printed Underwear Seamless Printed Bralette for Sex Naughty Play $14.99 [button] B09S3BN15C [button_] Mens Linen Shirt,Men's Striped Shirts Casual Short Sleeve Button Down Shirts Regular Fit Hawaiian Shirts Beach Tees Tops $3.78 to $11.38 [button] B09R9YCM6R [button_] XXBR Summer T-shirts for Mens, Soldier Short Sleeve 3D Street Vintage Printed Shirt Slim Fit Muscle Casual Tee Tops $8.98 to $11.99 [button] B09KP78G37 [button_] Women Faux Fur Lined Jacket Coat Winter Warm Thick Fleece Outwear Trench Zipper Plus Size Long Sleeve Plush Overcoat $47.41 to $59.07 [button] B00O30JLDK [button_] Legendary Whitetails Men's Non-Typical Long Sleeve T-Shirt $10.52 to $40.5 [button] B09Q67H373 [button_] JSPOYOU 2022 Newly T-Shirt for Mens Funny 3D Graphics Pattern Crewneck Short Sleeve Tees Big and Tall Summer Casual Comfy Top $1.99 to $8.99 ', Action 2: 'click[b09nd8p2qr]'] [Observation 3: 'Instruction: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Back to Search [button_] [button] < Prev [button_] size [button] small [button_] [button] medium [button_] [button] large [button_] [button] x-large [button_] [button] xx-large [button_] color [button] multi 1 [button_] [button] multi 10 [button_] [button] multi 2 [button_] [button] multi 3 [button_] [button] multi 4 [button_] [button] multi 5 [button_] [button] multi 6 [button_] [button] multi 7 [button_] [button] multi 8 [button_] [button] multi 9 [button_] InterestPrint Deer and Stars Men's Loungewear Pajama Sets - Long Sleeve Tee and Jogger Pant Price: $43.59 Rating: N.A. [button] Description [button_] [button] Features [button_] [button] Reviews [button_] [button] Buy Now [button_] ', Action 3: 'click[multi 7]'] You are now at step 4 and your current observation is: You have clicked multi 7. Instruction: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Back to Search [button_] [button] < Prev [button_] size [button] small [button_] [button] medium [button_] [button] large [button_] [button] x-large [button_] [button] xx-large [button_] color [button] multi 1 [button_] [button] multi 10 [button_] [button] multi 2 [button_] [button] multi 3 [button_] [button] multi 4 [button_] [button] multi 5 [button_] [button] multi 6 [button_] [clicked button] multi 7 [clicked button_] [button] multi 8 [button_] [button] multi 9 [button_] InterestPrint Deer and Stars Men's Loungewear Pajama Sets - Long Sleeve Tee and Jogger Pant Price: $43.59 Rating: N.A. [button] Description [button_] [button] Features [button_] [button] Reviews [button_] [button] Buy Now [button_] . Your admissible actions of the current situation are: ['click[back to search]', 'click[< prev]', 'click[description]', 'click[features]', 'click[reviews]', 'click[buy now]', 'click[small]', 'click[medium]', 'click[large]', 'click[x-large]', 'click[xx-large]', 'click[multi 1]', 'click[multi 10]', 'click[multi 2]', 'click[multi 3]', 'click[multi 4]', 'click[multi 5]', 'click[multi 6]', 'click[multi 7]', 'click[multi 8]', 'click[multi 9]',]. 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> To find men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7 and size: large, and price lower than 80.00 dollars, I need to search for products that meet these criteria. I have already taken the first three steps to narrow down the search, but I need to continue searching to find the specific product I am looking for. </think> <action>click[large]</action>

Prompt (WebShop, Step 5)

You are an expert autonomous agent operating in the WebShop e-commerce environment. Your task is to: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 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: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Back to Search [button_] [button] < Prev [button_] size [button] small [button_] [button] medium [button_] [button] large [button_] [button] x-large [button_] [button] xx-large [button_] color [button] multi 1 [button_] [button] multi 10 [button_] [button] multi 2 [button_] [button] multi 3 [button_] [button] multi 4 [button_] [button] multi 5 [button_] [button] multi 6 [button_] [button] multi 7 [button_] [button] multi 8 [button_] [button] multi 9 [button_] InterestPrint Deer and Stars Men's Loungewear Pajama Sets - Long Sleeve Tee and Jogger Pant Price: $43.59 Rating: N.A. [button] Description [button_] [button] Features [button_] [button] Reviews [button_] [button] Buy Now [button_] ', Action 3: 'click[multi 7]'] [Observation 4: 'You have clicked multi 7. Instruction: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Back to Search [button_] [button] < Prev [button_] size [button] small [button_] [button] medium [button_] [button] large [button_] [button] x-large [button_] [button] xx-large [button_] color [button] multi 1 [button_] [button] multi 10 [button_] [button] multi 2 [button_] [button] multi 3 [button_] [button] multi 4 [button_] [button] multi 5 [button_] [button] multi 6 [button_] [clicked button] multi 7 [clicked button_] [button] multi 8 [button_] [button] multi 9 [button_] InterestPrint Deer and Stars Men's Loungewear Pajama Sets - Long Sleeve Tee and Jogger Pant Price: $43.59 Rating: N.A. [button] Description [button_] [button] Features [button_] [button] Reviews [button_] [button] Buy Now [button_] ', Action 4: 'click[large]'] You are now at step 5 and your current observation is: You have clicked multi 7. You have clicked large. Instruction: Find me men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars [button] Back to Search [button_] [button] < Prev [button_] size [button] small [button_] [button] medium [button_] [clicked button] large [clicked button_] [button] x-large [button_] [button] xx-large [button_] color [button] multi 1 [button_] [button] multi 10 [button_] [button] multi 2 [button_] [button] multi 3 [button_] [button] multi 4 [button_] [button] multi 5 [button_] [button] multi 6 [button_] [clicked button] multi 7 [clicked button_] [button] multi 8 [button_] [button] multi 9 [button_] InterestPrint Deer and Stars Men's Loungewear Pajama Sets - Long Sleeve Tee and Jogger Pant Price: $43.59 Rating: N.A. [button] Description [button_] [button] Features [button_] [button] Reviews [button_] [button] Buy Now [button_] . Your admissible actions of the current situation are: ['click[back to search]', 'click[< prev]', 'click[description]', 'click[features]', 'click[reviews]', 'click[buy now]', 'click[small]', 'click[medium]', 'click[large]', 'click[x-large]', 'click[xx-large]', 'click[multi 1]', 'click[multi 10]', 'click[multi 2]', 'click[multi 3]', 'click[multi 4]', 'click[multi 5]', 'click[multi 6]', 'click[multi 7]', 'click[multi 8]', 'click[multi 9]',]. 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> To find men's sleep & lounge with long sleeve, elastic waistband for daily wear with color: multi 7, and size: large, and price lower than 80.00 dollars, I need to search for products that match these criteria. </think> <action>click[buy now]</action>

Outcome: Strict success. Task Score: 100.