LEARNING FROM ALL-FAILED GROUPS VIA FIRST-VISIT COVERAGE IN LONG-HORIZON AGENTIC TASKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Group-based policy optimization has been increasingly used to train large language model (LLM) agents from sparse outcome rewards by comparing trajectories or steps within a group. However, on difficult long-horizon agentic tasks, we found this comparison suffers from a sampling imbalance: *repeated or low-effect actions dominate the high-probability region of the policy*, leaving *useful state-changing actions under-sampled*. This imbalance produces many **all-failed rollout groups**, where the relative advantages within such a group provide no direction for updating the policy. Consequently, the sampling imbalance and the all-failed groups collapse into a self-perpetuating ***credit trap***: *failure-dominated sampling yields no learning signal, allowing repeated low-effect actions to persist.* To break this trap, we propose Coverage-conditioned Group Policy Optimization (CovGPO), which uses first-visit observation coverage only when all samples in a group receive zero outcome reward. Specifically, within such groups, CovGPO assigns higher relative advantages to state-exploring actions without corrupting the original reward signal, as reaching new observations is a prerequisite for task success. Experiments on two challenging agentic benchmarks, ALFWorld and WebShop, show that CovGPO consistently improves over group-based baselines, with particularly large gains on hard tasks.

1 INTRODUCTION

LLM agents (Achiam et al., 2023; Team et al., 2023; Yang et al., 2025; Team et al., 2025) are increasingly moving beyond single-turn response generation toward interactive problem solving in open-ended environments such as household tasks (Shridhar et al., 2020; Li et al., 2024), web navigation (Gou et al., 2025; Furuta et al., 2024), and tool use (Schick et al., 2023; Qian et al., 2026). These tasks require agents to plan, act, observe, and revise over many steps (Feng et al., 2025; Wang et al., 2023; 2024), while success is often revealed only through sparse outcome feedback. Such sparse supervision leaves most intermediate decisions without direct task feedback, making credit assignment over long trajectories difficult.

To train agents under such sparse outcome feedback, recent work has adopted group-based policy optimization (Shao et al., 2024; Yu et al., 2025; Kool et al., 2019), which samples multiple trajectories for the same task and updates the policy using their relative outcomes instead of a learned value function. At the trajectory level, GRPO compares the complete trajectories within each rollout group and assigns a single advantage to their constituent steps. To provide finer-grained credit, GiGPO (Feng et al., 2026) groups steps that share an anchor observation and produces step-varying advantages within the same trajectory, while HGPO (He et al., 2026) further groups steps by their shared interaction history. However, they all rely on the same assumption: *outcome variation among compared trajectories or steps*. When this variation vanishes, the comparison yields no signal.

This assumption is often reasonable in short-horizon or moderately difficult tasks (Liu et al., 2025; Yu et al., 2025; Wei et al., 2026), where some sampled trajectories succeed while others fail. However, in hard long-horizon environments (Shridhar et al., 2020; Yao et al., 2022a), many *all-failed rollout groups* occur: **every trajectory receives the same failure outcome**. In such groups, the policy exhibits a sampling imbalance: *high-probability actions tend to be low-effect while effective actions remain under-sampled* (Fig. 1). This leads to a *self-reinforcing credit trap*: failure-dominated

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.

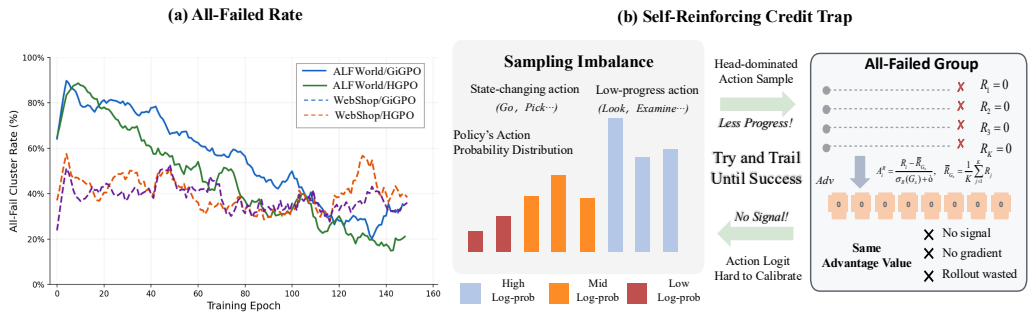

Figure 1: All-failed rate across training and credit trap. (a) Most rollout groups are all-failed during early training, yielding zero reward-based advantages. (b) The base policy concentrates probability mass on low-effect actions, leaving state-changing actions under-sampled. These two effects reinforce each other: imbalanced sampling produces more all-failed groups, while the resulting zero gradients prevent the policy from correcting the imbalance.

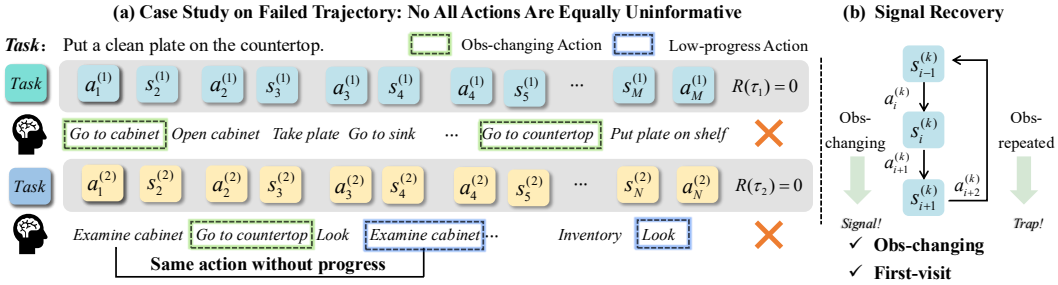

Figure 2: Not all failures are equal and progress signal. (a) Within the same all-failed group, some trajectories visit novel states while others stagnate on repeated no-change actions, revealing behavioral contrasts that reward-only algorithm discard. (b) We recover this signal via first-visit observation coverage, where each transition to a previously unseen observation contributes to coverage.

sampling produces all-failed groups, and because every trajectory shares the same outcome, the reward-based comparison provides no direction for correcting the imbalanced sampling.

To address this issue, recent research has shifted to learned progress models (Wang et al., 2025b; Xi et al., 2026; Chai et al., 2026), world-model guidance (Yu et al., 2026), and turn-level credit designs (Ma et al., 2026). However, these designs primarily depend on auxiliary supervision or learned models, instead of mining the intrinsic behavioral signals within the rollouts themselves. Therefore, an insightful question naturally arises: ***Can we recover an actionable signal directly from existing all-failed rollouts, without relying on auxiliary supervision or learned models?***

In this paper, we argue that the credit trap does not imply that failed rollouts are uninformative: equal failure outcomes do not imply equal behavior. Within the same all-failed group, one trajectory may repeatedly revisit the same observations, while another reaches more distinct observations before ultimately failing (Fig. 2 (a)). Although both receive the same terminal outcome, the latter exhibits more progress than the former. Existing methods fail to capture this contrast: GRPO assigns both trajectories the same zero advantage, while GiGPO and HGPO compare steps whose inherited returns are also identical, causing their step-level advantages to vanish as well.

Motivated by this insight, we propose Coverage-conditioned Group Policy Optimization (CovGPO), a fallback algorithm that derives an exploration signal from all-failed groups via **first-visit observation coverage**. Specifically, CovGPO identifies an interaction step as coverage-expanding if and only if it transitions to an observation not previously visited within the same trajectory. CovGPO credits transitions that expand observation coverage while assigning zero credit to actions that merely revisit or preserve existing states, as reaching novel observations is a necessary prerequisite for task success. Crucially, CovGPO applies this coverage credit *exclusively* as a fallback mechanism within all-failed groups, reverting strictly to the original reward-based advantage whenever outcome variance exists. This avoids rewarding aimless exploration over concise task completion. In this way, CovGPO breaks the self-reinforcing credit trap: within all-failed groups, trajectories

reaching new observations receive positive advantages, promoting the coverage-expanding actions the base policy was under-sampling. Across two challenging agentic benchmarks, ALFWorld and WebShop, using Qwen2.5-1.5B/7B-Instruct models, CovGPO consistently boosts strong existing baselines. Our contributions are summarized as follows:

- We identify the *all-failed credit trap*: imbalanced sampling produces all-failed rollout groups whose identical outcomes prevent reward-based algorithm from correcting the imbalance, despite the behavioral contrast within these groups.
- We propose CovGPO, a conditional fallback algorithm that recovers credit in all-failed groups via first-visit observation coverage, a necessary precondition for task success, and reverts exactly to the base advantage when outcome variation returns.
- On ALFWorld and WebShop with Qwen2.5-1.5B/7B-Instruct, CovGPO consistently boosts strong group-based baselines under the same computational constraints.

2 Preliminaries and Problem Formulation

Multi-Turn Agentic Tasks. Let $x \sim p(X)$ be a task instruction and π_θ an LLM-based policy. At each step t, the policy produces $a_t \sim \pi_\theta(\cdot \mid h_t)$ conditioned on the interaction history $h_t = (x, o_1, a_1, \ldots, o_t)$, and the environment returns the next observation o_{t+1}. The resulting trajectory is $\tau = (o_1, a_1, \ldots, o_T, a_T)$, where T is bounded by the interaction budget. Environments provide only a delayed terminal reward $R(\tau) \in [0, 1]$, evaluated via binary completion in ALFWorld and fractional matching in WebShop, leaving intermediate transitions entirely unrewarded.

Group-Based Reinforcement Learning. Group-based RL methods such as GRPO (Shao et al., 2024) estimate advantages critic-free from intra-group statistics. For task x, a group $G_x = \{\tau_i\}_{i=1}^K$ of K trajectories is sampled from π_θ, each receiving an outcome reward $R_i = R(\tau_i)$. The trajectory-level group-relative advantage is

$$A_i^R = \frac{R_i - \bar{R}_{G_x}}{\sigma_R(G_x) + \epsilon}, \quad \bar{R}_{G_x} = \frac{1}{K}\sum_{j=1}^{K} R_j, \tag{1}$$

and every step in τ_i shares A_i^R. For finer-grained credit, recent variants form step-level groups: GiGPO (Feng et al., 2026) collects steps across G_x that share the same environment state s into $G^S(s)$ and normalizes their discounted returns $R_{i,t}^S = \gamma^{T_i - t} R_i$ within it, while HGPO (He et al., 2026) groups steps by shared interaction history prefixes instead of a single anchor state.

$$A_{i,t}^S = \frac{R_{i,t}^S - \bar{R}_{G^S(s)}}{\sigma_R(G^S(s)) + \epsilon}, \tag{2}$$

The policy is updated via a clipped surrogate objective with KL regularization against a reference π_{ref}. Regardless of granularity, these algorithms work only when *within-group reward variation* exists.

All-Failed Groups and the Credit Trap. In hard long-horizon tasks, success is rare early in training, and a rollout group may contain no successful trajectory. We call such G_x an *all-failed rollout group*:

$$G_x \text{ is all-failed} \iff R_i = 0, \forall \tau_i \in G_x \iff \bar{R}_{G_x} = 0, \tag{3}$$

where the second equivalence uses $R_i \geq 0$. The trajectory-level algorithm in Eq. 1 then degenerates:

$$\sigma_R(G_x) = 0 \Longrightarrow A_i^R = 0, \forall i \in 1, \ldots, K. \tag{4}$$

When a group is all-failed, within-group reward variance vanishes ($\sigma_R(G_x) = 0$), and every discounted step return collapses to zero ($R_{i,t}^S = \gamma^{T_i - t} \cdot 0 = 0$). Across all grouping granularities, such groups yield no relative gradient signal, as identified by the unified condition in Eq. 3. Unlike all-successful groups with positive rewards or fractional-reward settings with non-zero partial returns, all-failed groups offer strictly zero feedback across all members.

3 Training Agents with CovGPO for Long-Horizon Agentic Tasks

A simple insight behind our method is that ***the behavioral contrast among failed trajectories is observable and can serve as a fallback learning signal when the reward degenerates***. We first

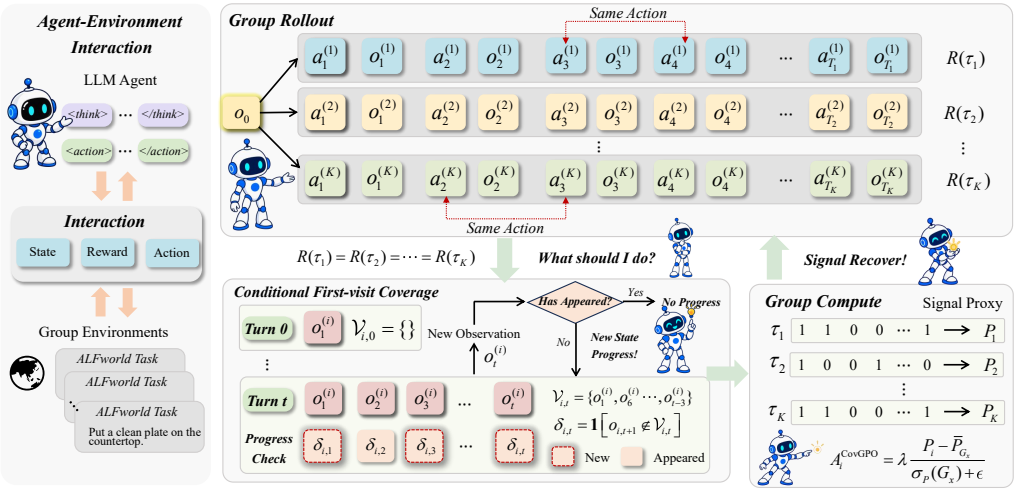

Figure 3: Overview of CovGPO. For each task, the agent samples a group of K rollouts from the same initial state and prompt. When the group is all-failed, CovGPO scores each trajectory by its first-visit observation coverage, and normalizes the scores within the group into a fallback advantage.

analyze the impact of this degeneracy (Section 3.1), then operationalize the repair through two components, illustrated in Figure 3: a *first-visit coverage score* (Section 3.2) that quantifies the contrast, and a *conditional switch* (Section 3.3) that uses this score only when the reward-based algorithm has degenerated. All proofs are deferred to Appendix B.

3.1 Why the Trap Is Exact

Equation 4 shows that all-failed groups yield identically zero advantages, where within-group credit signals do not merely weaken under sparse rewards but vanish entirely.

Proposition 1 (Exact degeneracy and fixed-policy sampling)**.** *Let $\mathcal{C}_H$ be the depth-H chain: states $s_0, \ldots, s_H$, actions $a^+ : s_j \mapsto s_{j+1}$ and $a^- : s_j \mapsto s_j$, horizon $T = H$, reward $R(\tau) = \mathbf{1}[s_H \text{ reached}]$, and initial policy $\pi_{\theta_0} = \pi_{\text{ref}}$ with $\pi_{\theta_0}(a^+ \mid s_j) = \varepsilon$ for all j. Whenever no rollout in a group reaches s_H, the reward advantages in Eqs. 1–2 are zero at every comparison granularity. If groups of K independent rollouts are sampled independently from the fixed policy π_{θ_0}, the expected number of groups until the first successful rollout is $[1-(1-\varepsilon^H)^K]^{-1} \geq \varepsilon^{-H}/K$.*

The zero is exact, not merely small: the group mean under a uniform outcome equals the observed return exactly, so the within-group reward signal vanishes identically. This identifies a missing source of signal, even though updates from other groups can still improve the policy.

The reward-only comparison leaves behavioral differences within all-failed groups unused. These already collected groups offer an opportunity to recover a signal without additional rollouts. *First,* recovering credit from *already collected* all-failed groups requires modifying the signal itself. Reshaping sampling distributions only affects future rollouts (Appendix D.2), leaving every member in the current degenerate group with zero credit. *Second,* the injected signal must distinguish behavior within the failed group. Restoring variation alone need not provide useful exploration, which motivates the coverage score below and the signal-substitution experiments in Section 4.3.

3.2 First-Visit Coverage Score

For the t-th transition in trajectory τ_i, we define

$$\delta_{i,t} = \mathbf{1}[o_{i,t+1} \notin \mathcal{V}_{i,t}], \qquad P_i = \frac{1}{T_i} \sum_{t=1}^{T_i} \delta_{i,t}, \tag{5}$$

where $\mathbf{1}[\cdot]$ is the indicator function, $\mathcal{V}_{i,t} = \{o_{i,1}, \ldots, o_{i,t}\}$ collects the observations already encountered in τ_i up to step t, and T_i is the length of τ_i, so that $0 \leq P_i \leq 1$. A step gets credit only when its resulting observation is new to the trajectory, which filters both low-effect transitions (the observation is unchanged) and cyclic revisitations (the observation returns to a seen state); a per-trajectory hash set adds $O(1)$ expected overhead per step.

Our contribution is not the score itself but the algorithm built around it: P enters the advantage only on groups where the reward signal has degenerated. *First*, $P(\tau) = (D(\tau) - 1)/T$, where $D(\tau)$ counts distinct observations (Lemma 1): at equal length P orders trajectories exactly by observation coverage, and repetitions and cycles earn no additional credit. For unequal lengths, P compares coverage rates instead of distinct-observation counts. *Second*, for binary outcomes, coverage certifies *necessity, not proximity*: writing m_x for the fewest distinct observations visited by any successful length-T trajectory, $P(\tau) < (m_x - 1)/T$ implies $R(\tau) = 0$. This necessary condition does not determine which failed trajectory is closer to success. We use P as an exploration-oriented ordering and assess its usefulness empirically.

That distance bounds the work still required, not the work still expected (Appendix B.3): a trajectory can satisfy the coverage requirement while standing in the wrong place holding the wrong object. We test this stronger hypothesis directly in Section 4.3 by resuming failed rollouts from their exact terminal states, and find only a small, statistically insignificant difference in favour of the higher-coverage trajectory (Appendix B.4 formalizes what would follow if a genuine ordering existed). In any case, P acts only on all-failed groups: since it tracks state coverage instead of task value, the conditional switch below keeps it from competing with the reward once outcome contrast returns.

3.3 Conditional Advantage Algorithm

An always-on coverage signal may conflict with the task reward once outcome variation is available: a short, efficient success can score lower on coverage than a long, wandering failure (Appendix B.10), so the auxiliary term partially offsets the reward-induced gradient (Tab. 2). Conversely, discarding coverage in all-failed groups wastes the only within-group contrast available. CovGPO therefore switches *conditionally*, on the observable within-group variance:

$$A_i^{\mathrm{CovGPO}} = \begin{cases} \lambda \dfrac{P_i - \bar{P}_{G_x}}{\sigma_P(G_x) + \epsilon}, & \text{if } \bar{R}_{G_x} = 0 \text{ and } \sigma_P(G_x) \geq \tau_P, \\ A_i^R = \dfrac{R_i - \bar{R}_{G_x}}{\sigma_R(G_x) + \epsilon}, & \text{otherwise,} \end{cases} \tag{6}$$

where $\bar{P}_{G_x}$ and $\sigma_P(G_x)$ mirror $\bar{R}_{G_x}$ and $\sigma_R(G_x)$ in Eq. 1, and $\lambda > 0$ scales the fallback ceiling. The trigger consists of a main condition and a safety check. The main condition holds exactly when every trajectory in the group receives zero reward, i.e., when the base advantage is identically zero (Eq. 4): this matches all-failed groups in ALFWorld and, under WebShop's fractional reward, further requires no partial match. The safety check $\sigma_P(G_x) \geq \tau_P$ disables the fallback when coverage is identical, avoiding division by near-zero variance.

For trajectory-level base algorithms, P_i directly scores each rollout. For step-level baselines (e.g., GiGPO and HGPO), constituent steps inherit their parent trajectory's score (Appendix C.3). In both cases, the fallback injects a coverage signal if and only if the base reward advantage vanishes, leaving informative groups completely intact.

Proposition 2 (Non-interference)**.** *For any policy π_θ, any threshold $\tau_P > 0$, and any group G_x: if G_x does not satisfy the fallback condition of Eq. 6, then $A_i^{\mathrm{CovGPO}} = A_i^R$ for all i; if it does, then $A_i^R = 0$ for all i. Consequently, every loss sample that depends on no triggered comparison keeps both its value and its gradient (Appendix B.5).*

CovGPO is therefore a *strict extension* of the base algorithm, modifying the update only where the base method provably provides no signal, at any grouping granularity. The statement concerns the algorithm at a fixed θ; because the two runs subsequently visit different data, it does not claim that the two *training trajectories* agree on reward-informative groups.

Within each triggered fallback group, the update applies a group-mean baselined step on coverage P among failed trajectories. Although Lemma 4 establishes the $(K-1)/K$ relation for the idealized, unclipped statistic before dispersion filtering, the deployed update serves as a regularized surrogate instead of an exact gradient identity. Crucially, the fallback update satisfies two analytical properties. The coverage score-function step is bounded as shown in Proposition 5 and evaluates to zero whenever coverage dispersion drops below the gating threshold. Furthermore, for K i.i.d. trajectory rollouts, the trigger probability is at most $(1 - p_+(x))^K$ with $p_+(x) = \mathbb{P}(R > 0 \mid x)$ (Remark 1).

Finally, CovGPO integrates into any group-based framework by replacing the reward-only advantage with A_i^{CovGPO} in a clipped surrogate objective:

$$\mathcal{J}_{\mathrm{CovGPO}}(\theta) = \mathbb{E}_{x,G_x}\Big[\frac{1}{K}\sum_{i=1}^{K}\frac{1}{T_i}\sum_{t=1}^{T_i}\min\big(\rho_{i,t}(\theta)\,A_i^{\mathrm{CovGPO}},\,\mathrm{clip}\big(\rho_{i,t}(\theta),\,1-\epsilon_c,\,1+\epsilon_c\big)A_i^{\mathrm{CovGPO}}\big)\Big] - \beta\,D_{\mathrm{KL}}(\pi_\theta\|\pi_{\mathrm{ref}}), \tag{7}$$

where $\rho_{i,t}(\theta) = \pi_\theta(a_{i,t} \mid h_{i,t})/\pi_{\theta_{\mathrm{old}}}(a_{i,t} \mid h_{i,t})$ is the importance ratio, ϵ_c is the clipping range, and β controls KL regularization.

4 Experiments

4.1 Experiment Setup

Benchmarks. We train and evaluate LLM agents on two challenging benchmarks: ALFWorld (Shridhar et al., 2020) and WebShop (Yao et al., 2022a). *ALFWorld* is an embodied text environment for multi-step decision-making. In each episode, the agent receives a textual goal and must accomplish it through multi-turn interaction spanning 20–50 steps, receiving only a binary outcome reward upon termination. It covers six categories of household activities: Pick & Place (Pick), Examine in Light (Look), Clean & Place (Clean), Heat & Place (Heat), Cool & Place (Cool), and Pick Two & Place (Pick2). *WebShop* is a web-based interactive shopping environment in which the agent must search, navigate, and purchase a suitable item on a simulated HTML website. *SokoBan* is a visual puzzle environment where a vision-language agent observes RGB images and must plan directional pushes to navigate boxes to target locations without entering irreversible deadlocks.

Baselines. We compare CovGPO against three groups of methods. (1) *Prompting agents*: the base Qwen2.5 models, React (Yao et al., 2022b) and Reflexion (Shinn et al., 2023), which guide multi-step behavior through in-context prompting without parameter updates. (2) *RL training methods*: PPO (Schulman et al., 2017), which requires an additional critic network; RLOO (Kool et al., 2019), a critic-free REINFORCE variant with a leave-one-out baseline; DAPO-MT, which adapts DAPO (Yu et al., 2025) to multi-turn rollouts by resampling until reward variance emerges—contrasting with our reuse of existing transitions; and the group-based methods GRPO (Shao et al., 2024), GiGPO (Feng et al., 2026), and HGPO (He et al., 2026), where the latter is the state-of-the-art group-based algorithm for LLM agents and serves as our primary base algorithm. (3) *CovGPO-augmented methods*: since CovGPO modifies only the advantage computation, we instantiate it on top of three different base algorithms, GRPO+CovGPO, GiGPO+CovGPO and HGPO+CovGPO. Full training settings are provided in Appendix C.

Training details. We use Qwen2.5-1.5B/7B-Instruct, rollout groups of size $K{=}8$, and 150 training epochs. Across environments, model scales, and base algorithms, we fix the configured fallback ceiling at $\lambda_{\mathrm{aux}}{=}0.3$. We report success rate (%) on held-out test tasks; we report mean$_{\pm\mathrm{std}}$ with three random seeds. More details are provided in Appendix C.

4.2 Main Results

Table 1 summarizes the main results. Prompting-only agents, including strong closed-source models, still struggle on these long-horizon tasks, while RL training improves substantially over prompting. We highlight three observations: **(i) CovGPO improves group-based RL across scales and benchmarks.** At 1.5B it lifts GRPO/GiGPO/HGPO on ALFWorld ($72.8{\to}74.5$, $86.7{\to}92.2$, $92.8{\to}95.7$) and on WebShop success ($56.8{\to}60.9$, $65.0{\to}72.4$, $71.5{\to}75.7$); the same pattern holds at 7B, where GRPO and GiGPO rise on ALFWorld ($77.6{\to}79.7$, $90.8{\to}92.5$) and every base improves on WebShop success ($66.1{\to}68.2$, $72.8{\to}76.2$, $78.5{\to}79.7$). The only saturated setting is 7B HGPO, where CovGPO behaves as predicted: ALFWorld changes from 95.4 to 95.7 (±0.6), WebShop score remains 89.0, and success gains $1.2, 78, 5{\to}79.7$, all within seed variation. In this setting, the base policy rarely produces all-failed groups, so the fallback is seldom activated; on the remaining nondegenerate groups, CovGPO exactly recovers the base advantage by Proposition 2. **(ii) CovGPO's gains concentrate on hard subtasks.** Improvements are largest where the base policy almost never succeeds and all-failed groups dominate: at 1.5B, GiGPO+CovGPO lifts *Look*

Table 1: Test performance on ALFWorld and WebShop. For ALFWorld, we report the average success rate (%) for each subtask as well as the overall result; for WebShop, we report the average task score and the average success rate (%). Most results are averaged over 3 random seeds; the 1.5B GiGPO+CovGPO ALFWorld entry is the five-random-seed average of Table 15 (λ_{aux}=0.3). Seed protocols are specified in Appendix C.8. ↑ / ↓ give the change on the aggregate metrics relative to that base algorithm. **Bold** marks the best result within each model scale.

Method	ALFWorld							WebShop	
	Pick	Look	Clean	Heat	Cool	Pick2	All	Score	Succ.
Closed-Source Models									
GPT-4o	75.3	60.8	31.2	56.7	21.6	49.8	48.0	31.8	23.7
Gemini-2.5-Pro	92.8	63.3	62.1	69.0	26.6	58.7	60.3	42.5	35.9
Qwen2.5-1.5B-Instruct									
Qwen2.5	5.9	5.5	3.3	9.7	4.2	0.0	4.1	23.1	5.2
ReAct	17.4	20.5	15.7	6.2	7.7	2.0	12.8	40.1	11.3
Reflexion	35.3	22.2	21.7	13.6	19.4	3.7	21.8	55.8	21.9
PPO	64.8±3.5	40.5±6.9	57.1±4.9	60.6±6.6	46.4±4.0	47.4±1.9	54.4±3.1	73.8±3.0	51.5±2.9
RLOO	88.3±3.0	52.8±8.6	71.0±5.9	62.8±8.7	66.4±5.5	56.9±4.7	69.7±2.5	73.9±5.6	52.1±6.7
DAPO-MT	94.6±1.9	75.6±10.2	83.4±4.1	85.3±3.1	86.4±7.2	78.1±4.9	84.9±0.9	75.0±1.2	62.9±0.4
GRPO	85.3±1.5	53.7±8.0	84.5±6.8	78.2±7.9	59.7±5.0	53.5±5.6	72.8±3.6	75.8±3.5	56.8±3.8
GRPO + CovGPO	84.1±5.6	76.3±15.1	74.8±8.1	69.6±9.2	83.2±4.4	50.8±11.3	74.5±1.6 **↑1.7**	76.2±1.0 **↑0.4**	60.9±2.2 **↑4.1**
GiGPO	94.4±5.9	67.5±4.6	94.8±3.8	94.4±7.8	79.8±4.7	76.4±5.4	86.7±1.7	83.1±1.6	65.0±3.2
GiGPO + CovGPO	94.9±3.9	74.5±12.3	**98.4**±2.0	95.9±5.4	90.6±4.3	90.9±3.5	92.2±2.0 **↑5.5**	85.6±0.5 **↑2.5**	72.4±0.6 **↑7.4**
HGPO	97.8±0.3	85.4±5.9	95.0±3.1	**97.4**±1.7	87.6±2.6	88.0±6.0	92.8±1.1	85.6±2.9	71.5±4.0
HGPO + CovGPO	**100.0**±0.0	**91.3**±0.5	98.3±2.4	97.1±4.2	**92.3**±2.0	**93.5**±2.5	**95.7**±0.6 **↑2.9**	**87.5**±1.6 **↑1.9**	**75.7**±2.8 **↑4.2**
Qwen2.5-7B-Instruct									
Qwen2.5	33.4	21.6	19.3	6.9	2.8	3.2	14.8	26.4	7.8
ReAct	48.5	35.4	34.3	13.2	18.2	17.6	31.2	46.2	19.5
Reflexion	62.0	41.6	44.9	30.9	36.3	23.8	42.7	58.1	28.8
PPO	92.3±4.0	64.0±8.4	92.5±2.4	89.5±7.0	80.3±2.0	68.8±8.3	80.4±2.7	81.4±3.1	68.7±5.1
RLOO	87.6±4.3	78.2±8.3	87.3±5.8	81.3±7.6	71.9±5.2	48.9±8.4	75.5±4.6	80.3±3.2	65.7±4.0
DAPO-MT	98.9±1.9	83.3±5.8	95.2±1.7	90.8±3.0	79.1±9.2	87.0±1.0	90.4±1.9	83.0±1.5	74.6±2.5
GRPO	90.8±5.1	66.1±6.7	89.3±5.4	74.7±6.9	72.5±5.4	64.7±7.3	77.6±5.2	79.3±2.8	66.1±3.7
GRPO + CovGPO	90.0±4.1	67.5±8.4	88.4±2.3	91.2±12.5	76.2±1.6	58.4±18.4	79.7±3.3 **↑2.1**	83.3±3.6 **↑4.0**	68.2±4.3 **↑2.1**
GiGPO	97.7±1.6	82.7±7.9	98.8±1.6	83.7±7.2	**89.3**±8.2	79.2±6.6	90.8±1.3	84.4±2.9	72.8±3.2
GiGPO + CovGPO	**100.0**±0.0	**97.2**±4.8	91.8±0.4	83.4±3.0	87.1±3.3	91.0±4.7	92.5±0.9 **↑1.7**	85.9±0.2 **↑1.5**	76.2±1.1 **↑3.4**
HGPO	98.9±1.2	88.8±1.6	**98.9**±0.4	**95.3**±6.0	86.8±1.3	**98.1**±1.2	95.4±0.6	**89.0**±1.0	78.5±1.4
HGPO + CovGPO	**100.0**±0.0	95.5±6.4	96.7±0.0	92.5±2.3	88.0±4.2	97.6±3.4	**95.7**±0.6 **↑0.3**	**89.0**±1.6	**79.7**±1.4 **↑1.2**

Table 2: Ablation on conditional gating (w/o Conditional = always-on coverage) on ALFWorld and WebShop (Qwen2.5-1.5B, GRPO and GiGPO bases, 150 epochs). **Bold** marks the best per column.

Method	ALFWorld							WebShop	
	Pick	Look	Clean	Heat	Cool	Pick2	All	Score	Succ.
GRPO	**85.3**±1.5	53.7±8.0	**84.5**±6.8	**78.2**±7.9	59.7±5.0	**53.5**±5.6	72.8±3.6	75.8±3.5	56.8±3.8
Always-on coverage	77.5±7.4	63.1±12.4	78.3±5.0	77.6±13.5	69.3±11.1	47.4±10.3	68.3±2.7	73.0±4.7	58.6±4.1
CovGPO (conditional)	84.1±5.6	**76.3**±15.1	74.8±8.1	69.6±9.2	**83.2**±4.4	50.8±11.3	**74.5**±1.6	**76.2**±1.0	**60.9**±2.2
GiGPO	94.4±5.9	67.5±4.6	94.8±3.8	94.4±7.8	79.8±4.7	76.4±5.4	86.7±1.7	83.1±1.6	65.0±3.2
Always-on coverage	**99.2**±1.6	**78.9**±12.1	95.7±4.0	**98.8**±2.4	81.9±8.6	87.4±1.3	90.9±2.5	78.5±3.2	68.8±2.9
CovGPO (conditional)	94.9±3.9	74.5±12.3	**98.4**±2.0	95.9±5.4	**90.6**±4.3	**90.9**±3.5	**92.2**±2.0	**85.6**±0.5	**72.4**±0.6

67.5→74.5, *Cool* 79.8→90.6, *Pick2* 76.4→90.9, while the easy *Pick* is saturated. This suggests that failed trajectories are not equally wrong, and CovGPO recovers the hidden contrast precisely in the all-failed regime. **(iii) CovGPO achieves sample-efficient gains without rollout overhead.** CovGPO reuses existing trajectories to resolve all-failed groups with zero extra environment interactions. In contrast, while DAPO-MT improves performance by repeatedly resampling rollouts until reward variance emerges, it incurs prohibitive generation overhead (10.5 h at 1.5B and 43.6 h at 7B). **(iv) CovGPO is compatible with both trajectory- and step-level algorithms.** Adding CovGPO increases aggregate ALFWorld success for GRPO, GiGPO, and HGPO at 1.5B (72.8→74.5, 86.7→92.2, and 92.8→95.7). This supports portability across grouping granularities. **(vi) Extension to visual environments.** CovGPO also extends to multimodal settings: an exploratory visual study on SokoBan (Table 3) improves GRPO, GiGPO, and HGPO by +6.3%, +2.0%, and +3.2%, respectively (details in Appendix E.1.1).

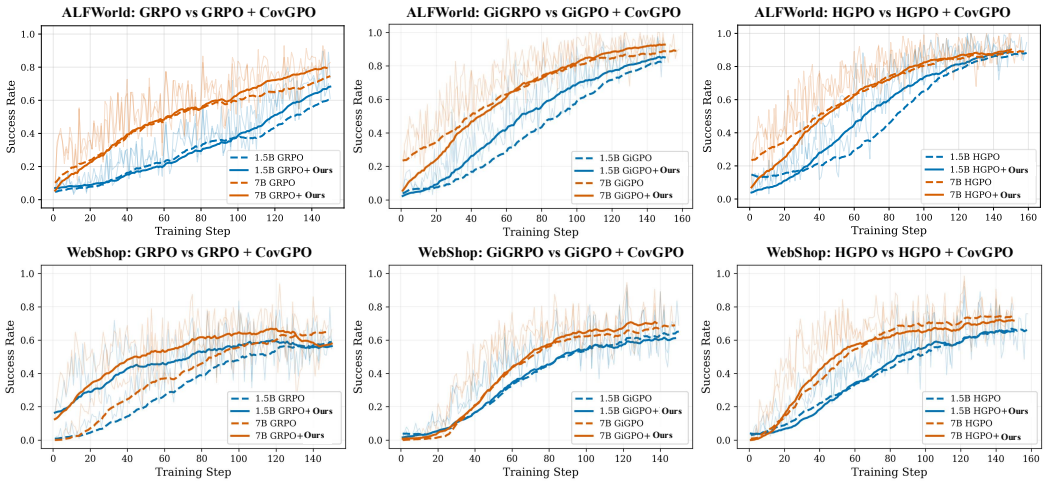

Figure 4: Training dynamics of CovGPO and its baseline on ALFWorld and WebShop. The lighter curves show the original curves, while the darker curves correspond to exponential moving average (EMA) smoothing with decay $\alpha = 0.95$, highlighting the overall training trends.

Table 3: Test performance of VLM agents using Qwen2.5-VL-3B-Instruct on the interactive game environment SokoBan.

Training	Vanilla	+CovGPO	Δ
Prompting	11.7	–	–
GRPO	67.1	**73.4**$\pm$3.2	**+6.3**
GiGPO	76.9	**78.9**$\pm$2.7	**+2.0**
HGPO	81.2	**84.4**$\pm$2.5	**+3.2**

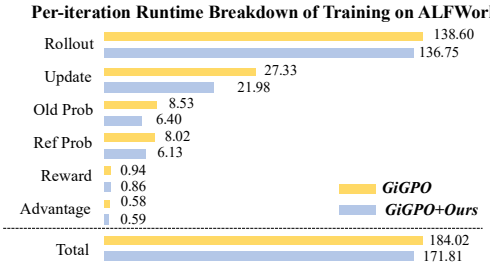

Figure 5: Training-time(s) delta on four GPUs.

Ablation Study. Always-on coverage (w/o Conditional) forfeits the non-interference guarantee and underperforms conditional CovGPO on average. The gap is largest for weaker base algorithms: it falls below the base algorithm itself on GRPO–ALFWorld (68.3 vs. 72.8), stays within seed variation of conditional CovGPO on GiGPO–ALFWorld, where the base is already strong (90.9 ± 2.5 vs. 92.2 ± 2.0), and is lower on both metrics on GiGPO–WebShop ($78.5/68.8$ vs. $85.6/72.4$). Gating is thus necessary for the exact guarantee .(Appendix B.10).

Training dynamics. Figure 4 plots success rate against training epochs, comparing each CovGPO-augmented method with its baseline under matched rollouts, hyperparameters, and seeds. On 1.5B ALFWorld, GiGPO+CovGPO reaches 49.2% success at epoch 60 versus GiGPO's 32.8%, and finishes at 91.4% in this single-seed trace (Appendix E.3). From epochs 10 to 150, its zero-score step-group fraction falls from 92.0% to 43.6%, while fallback use falls from 29.4% to 4.7% (Table 16). Intervention thus decreases without an epoch schedule; the rates differ because only coverage-diverse zero-score groups trigger the fallback.

Computational Budget. CovGPO adds an expected $O(1)$ hash-set lookup per step and scalar group statistics, with no learned components. In the timing audit (Figure 5), advantage computation is nearly unchanged (0.587 vs. 0.585 s); total iteration time is 184.0 vs. 171.8 s. These measurements show negligible added cost, not an intrinsic speedup.

4.3 Why CovGPO Works

We examine what the fallback signal contributes, how it changes action preferences, and which design choices matter. The analyses use Qwen2.5-1.5B-Instruct with GiGPO on ALFWorld; Figure 6 organizes the diagnostics, with protocols and corrected statistics in Appendices D.3 and E.4.

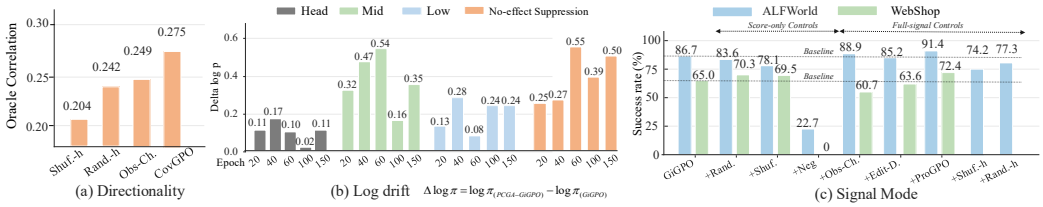

Figure 6: Policy-preference diagnostics and fallback-signal ablations on ALFWorld with Qwen2.5-1.5B-Instruct. (a) Correlation between policy log-probability changes and oracle-labeled task progress on fixed action records. (b) CovGPO suppresses low-effect actions and shifts probability mass toward under-sampled ones. (c) Task performance with different fallback signals.

CovGPO benefits from trajectory-aligned coverage credit. We replace coverage with shuffled or random scores in all-failed conditions: (i) *horizontal controls* operate *without* CovGPO's first-visit coverage signal entirely, replacing both the dispersion gate and fallback advantage ranking with permuted scores (*Shuffle-h*, 74.2%) or uniform noise (*Random-h*, 77.3%); (ii) *score-only controls* retain CovGPO's dispersion gate check but operate *without* trajectory-aligned coverage credit, permuting (*Shuffle*, 78.1%) or randomizing the injected scores (*Random*, 83.6%). Both underperform GiGPO (86.7%) and CovGPO (92.2%), demonstrating that neither non-zero variance nor dispersion gating alone suffices without trajectory-aligned coverage credit (Appendix E.4).

CovGPO steers policy sampling toward state-changing actions. Across fixed records, CovGPO consistently increases $\log \pi_{\text{CovGPO}} - \log \pi_{\text{GiGPO}}$ on mid-band state-changing transitions while penalizing low-effect actions. It outperforms *Shuffle-h* by $+0.032$ in oracle correlation and 2.357 nats in no-op suppression (Appendix D.3). Crucially, *Obs-Change* exhibits larger aggregate shifts but lower task success, demonstrating that redistribution magnitude alone does not drive gains.

CovGPO employs coverage as an exploration heuristic. First-visit coverage detects novel observations over repetition, but does not index failure quality. Resuming 246 paired failed rollouts from step 50 for 25 steps yields success rates of 4.47% and 3.25% for higher- and lower-coverage trajectories ($+1.22$ pp). Its partial correlation with oracle subgoal progress is marginal (Appendix E.2). These diagnostics do not establish coverage as a reliable continuation-value proxy; CovGPO thus uses it solely to guide exploration when reward-based comparisons offer no distinction.

Hyperparameter sensitivity. CovGPO's fallback ceiling λ_{aux} is fixed at 0.3 across environments. Sweeping $\lambda_{\text{aux}} \in \{0.1, 0.3, 0.5, 0.7, 1.0\}$ on ALFWorld (Table 15) shows that 0.3 achieves 92.2% success, within 2.3pp of the single-seed peak (0.1: 94.5%) and well above 0.5 (78.3%) and 0.7 (83.6%), confirming $\lambda_{\text{aux}} = 0.3$ as a robust default.

5 Conclusion

In this work, we identify the credit trap in group-based RL for multi-turn LLM agents, where uninformative zero-reward rollout groups yield degenerate relative advantages despite exhibiting distinct interaction behaviors. To recover this latent contrast, we propose Coverage-conditioned Group Policy Optimization (CovGPO), a lightweight, model-free framework that ranks trajectories within all-failed comparison groups via *first-visit observation coverage*. Computed directly from existing interactions without extra rollouts, learned reward models, or intermediate supervision, CovGPO seamlessly integrates with trajectory-, state-, and history-level baselines (GRPO, GiGPO, and HGPO). Theoretically, we prove that our gated fallback strictly preserves standard reward-relative policy gradients whenever reward variance exists. Evaluations across ALFWorld and WebShop show consistent gains across two model scales (Qwen2.5-1.5B/7B) and three base algorithms, suppressing repetitive, state-invariant actions. Crucially, our findings deliver a broader insight for agentic RL: ***an auxiliary behavioral statistic can provide powerful exploratory credit without serving as a calibrated value estimator or success proxy***. CovGPO demonstrates that structured failures offer a low-cost, readily accessible source of conditional credit, opening promising avenues for representation-level coverage metrics and adaptive disengagement in long-horizon agent learning.

AI USE STATEMENT

In this work, we used generative AI tools for assisting in the implementation of baseline and evaluation methods. We have not used generative AI tools for developing theoretical models, formulating mathematical claims, or assisting with proofs, nor for generating synthetic datasets or formulating core research hypotheses, and other required disclosure tasks (such as translation, dataset reformatting, or qualitative analysis) are not applicable to this work. Additionally, we used generative AI tools for editing the manuscript to improve readability and language polishing, restructuring sections, and writing auxiliary scripting code. We have thoroughly reviewed all AI-assisted work: all theoretical frameworks, propositions, and proofs were independently conceived and derived by the authors without AI assistance, and all algorithm implementations and experimental scripts were manually verified and verified via execution tests by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

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

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

Udbhav Bamba, Minghao Fang, Yifan Yu, Haizhong Zheng, and Fan Lai. Xrpo: Pushing the limits of grpo with targeted exploration and exploitation. *arXiv preprint arXiv:2510.06672*, 2025.

Guangfeng Cai, Kaibing Yang, Shuo He, Yu Li, Shengtian Yang, Jiaqi Lv, and Lei Feng. Beyond next-observation prediction: Agent-authored world modeling for sequential decision making. *arXiv preprint arXiv:2606.25421*, 2026.

Huacan Chai, Zijie Cao, Maolin Ran, Yingxuan Yang, Jianghao Lin, Xin Peng, Hairui Wang, Renjie Ding, Ziyu Wan, Muning Wen, et al. Progra: Progress-aware reinforcement learning for multi-turn function calling. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 6519–6535, 2026.

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 *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL 2026)*, pp. 285–307, 2026.

Lang Feng, Weihao Tan, Zhiyi Lyu, Longtao Zheng, Haiyang Xu, Ming Yan, Fei Huang, and Bo An. Towards efficient online tuning of vlm agents via counterfactual soft reinforcement learning. *arXiv preprint arXiv:2505.03792*, 2025.

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

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

Boyu Gou, Demi Ruohan Wang, Boyuan Zheng, Yanan Xie, Cheng Chang, Yiheng Shu, Huan Sun, and Yu Su. Navigating the digital world as humans do: Universal visual grounding for gui agents. In *International Conference on Learning Representations*, volume 2025, pp. 30851–30883, 2025.

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

Zhenyu Hou, Yujiang Li, Jie Tang, and Yuxiao Dong. Single-rollout asynchronous optimization for agentic reinforcement learning. *arXiv preprint arXiv:2607.07508*, 2026.

Wouter Kool, Herke Van Hoof, and Max Welling. Buy 4 reinforce samples, get a baseline for free! 2019.

Thanh-Long V Le, Myeongho Jeon, Kim Vu, Viet Lai, and Eunho Yang. No prompt left behind: Exploiting zero-variance prompts in llm reinforcement learning via entropy-guided advantage shaping. *arXiv preprint arXiv:2509.21880*, 2025.

Jiazheng Li, Yawei Wang, Qiaojing Yan, Yijun Tian, Zhichao Xu, Huan Song, Panpan Xu, and Lin Lee Cheong. Salt: Step-level advantage assignment for long-horizon agents via trajectory graph. In *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 4709–4725, 2026.

Manling Li, Shiyu Zhao, Qineng Wang, Kangrui Wang, Yu Zhou, Sanjana Srivastava, Cem Gokmen, Tony Lee, Li E Li, Ruohan Zhang, et al. Embodied agent interface: Benchmarking llms for embodied decision making. *Advances in Neural Information Processing Systems*, 37:100428–100534, 2024.

Xiaoqian Liu, Ke Wang, Yuchuan Wu, Fei Huang, Yongbin Li, Jianbin Jiao, and Junge Zhang. Agentic reinforcement learning with implicit step rewards. In *International Conference on Learning Representations*, volume 2026, pp. 129271–129291, 2026.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

Shichao Ma, Zhiyuan Ma, Ming Yang, Xiaofan Li, Xing Wu, Jintao Du, Yu Cheng, Weiqiang Wang, Qiliang Liu, Zhengyang Zhou, et al. Tspo: Breaking the double homogenization dilemma in multi-turn search policy optimization. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 28607–28623, 2026.

Gongrui Nan, Siye Chen, Jing Huang, Mengyu Lu, Dexun Wang, Chunmei Xie, Weiqi Xiong, Xianzhou Zeng, Qixuan Zhou, Yadong Li, et al. Ngrpo: Negative-enhanced group relative policy optimization. *arXiv preprint arXiv:2509.18851*, 2025.

Cheng Qian, Emre Can Acikgoz, Qi He, Hongru Wang, Xiusi Chen, Dilek Hakkani-Tur, Gokhan Tur, and Heng Ji. Toolrl: Reward is all tool learning needs. *Advances in Neural Information Processing Systems*, 38:105523–105553, 2026.

Roberta Raileanu and Tim Rocktäschel. Ride: Rewarding impact-driven exploration for procedurally-generated environments. *arXiv preprint arXiv:2002.12292*, 2020.

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

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

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

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

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

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

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

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

Guoqing Wang, Sunhao Dai, Guangze Ye, Zeyu Gan, Wei Yao, Yong Deng, Xiaofeng Wu, and Zhenzhe Ying. Information gain-based policy optimization: A simple and effective approach for multi-turn llm agents. *arXiv preprint arXiv:2510.14967*, 2025a.

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*, 2025b.

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*, 2025c.

Junyang Wang, Haiyang Xu, Haitao Jia, Xi Zhang, Ming Yan, Weizhou Shen, Ji Zhang, Fei Huang, and Jitao Sang. Mobile-agent-v2: Mobile device operation assistant with effective navigation via multi-agent collaboration. *Advances in Neural Information Processing Systems*, 37:2686–2710, 2024.

Yuxiang Wei, Olivier Duchenne, Jade Copet, Quentin Carbonneaux, Lingming Zhang, Daniel Fried, Gabriel Synnaeve, Rishabh Singh, and Sida Wang. Swe-rl: Advancing llm reasoning via reinforcement learning on open software evolution. *Advances in Neural Information Processing Systems*, 38:78500–78525, 2026.

Zhiheng Xi, Chenyang Liao, Guanyu Li, Zhihao Zhang, Wenxiang Chen, Binghai Wang, Senjie Jin, Yuhao Zhou, Jian Guan, Wei Wu, et al. Agentprm: Process reward models for llm agents via step-wise promise and progress. In *Proceedings of the ACM Web Conference 2026*, pp. 4184–4195, 2026.

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

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

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

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2025.

Xiao Yu, Baolin Peng, Ruize Xu, Yelong Shen, Pengcheng He, Suman Nath, Nikhil Singh, Jiangfeng Gao, and Zhou Yu. Reinforcement world model learning for llm-based agents. *arXiv preprint arXiv:2602.05842*, 2026.

Xichen Zhang, Sitong Wu, Yinghao Zhu, Haoru Tan, Shaozuo Yu, Ziyi He, and Jiaya Jia. Scaf-grpo: Scaffolded group relative policy optimization for enhancing llm reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 131946–131974, 2026.

Yifei Zhou, Andrea Zanette, Jiayi Pan, Sergey Levine, and Aviral Kumar. Archer: Training language model agents via hierarchical multi-turn rl. *arXiv preprint arXiv:2402.19446*, 2024.

Supplementary Material for "Learning from All-Failed Groups via First-Visit Coverage in Long-Horizon Agentic Tasks"

CONTENTS

1 Introduction **1**

2 Preliminaries and Problem Formulation **3**

3 Training Agents with CovGPO for Long-Horizon Agentic Tasks **3**

3.1 Why the Trap Is Exact 4

3.2 First-Visit Coverage Score 4

3.3 Conditional Advantage Algorithm 5

4 Experiments **6**

4.1 Experiment Setup 6

4.2 Main Results 6

4.3 Why CovGPO Works 8

5 Conclusion **9**

A Related Work **15**

B Proofs and Worked Example **16**

B.1 Proof of Proposition 1 16

B.2 Coverage Exactness 17

B.3 The Coverage Deficit Bounds the Remaining Work 18

B.4 Ordering under Coverage-Monotone Continuation 18

B.5 Proof of Proposition 2 19

B.6 The Unnormalized REINFORCE Identity and Groupwise Normalization 21

B.7 A Counterexample to Fallback-Only Escape Guarantees 23

B.8 A Static Norm Bound for the Gated Fallback 23

B.9 Disengagement Bounds: Two Conditions 24

B.10 Why the Fallback Must Be Conditional: Reward–Progress Conflict 25

B.11 A Worked all-failed Group: Instantiating the Main-Text Results 25

C Implementation Details **26**

C.1 Computing Details 27

C.2 Reference Advantage Computation 27

C.3 Deployed Per-Update Scaling and Conformance Boundary 27

C.4 Training Hyperparameters 30

C.5 ALFWorld 30
C.6 WebShop 30
C.7 Progress Measure across Environments 31
C.8 Evaluation and Reporting Protocol 31

D Sampling-Imbalance Audit and Sampler-Side Intervention **31**
D.1 Base-Policy Action Audit 32
D.2 Sampler-Side Intervention: Band-Conditioned Sampling 32
D.3 Action-Level Audit: Metric Definitions and Rationale 33

E Additional Results **34**
E.1 DAPO-MT Per-Category Results on ALFWorld 34
E.1.1 SokoBan: Implementation and Evaluation Linkage 34
E.2 What the Coverage Score Tracks: Protocol and Full Results 35
E.3 Per-Category Snapshots across Training Progress 37
E.4 Per-Category Breakdown of Signal-Analysis Variants 37
E.5 Novelty/Revisit Decomposition of the Obs-Change Anomaly 38
E.6 Random-Seed Reporting and Protocols 39
E.7 Fallback-Scale Sweep 39
E.8 Training Dynamics and all-failed Group Statistics 39
E.8.1 Zero-Score Group Rates across Training Progress 39
E.8.2 Fallback Frequency across Training Progress 39
E.8.3 Per-Category all-failed Composition 40
E.8.4 Fallback Runtime Diagnostics: Definitions and Additional Curves 40

F Case Study: all-failed Groups **41**
F.1 ALFWorld: Heat an Egg 41
F.2 WebShop: Searching for Pants 41

G Limitations and Broader Impact **42**

A Related Work

Group-Based Reinforcement Learning for Long-Horizon LLM Agents. Critic-free methods such as GRPO Shao et al. (2024) and RLOO (Ahmadian et al., 2024) estimate policy advantages from relative outcomes among multiple samples, reducing the memory and computation required by PPO-style actor–critic training (Schulman et al., 2017; Zhou et al., 2024; Hou et al., 2026). Their agentic extensions increasingly refine the comparison unit: GiGPO (Feng et al., 2026) combines trajectory-level and anchor-state groups, ProxMO (Fang et al., 2026) introduces difficulty-aware modulation and semantic soft aggregation, and HGPO (He et al., 2026) forms hierarchical groups to control historical-context inconsistency. SALT (Li et al., 2026) and GraphGPO (Cheng et al., 2026) exploit trajectory or state-transition graphs to propagate structural credit across related states. These methods improve the granularity or accuracy of credit assignment once a usable comparison signal exists. CovGPO addresses a complementary regime in which all sampled trajectories fail and the outcome statistics provide no signal, using first-visit observation coverage as a conditional fallback instead of introducing another grouping or value-estimation structure.

Learning from all-failed and outcome-degenerate groups. Group-based policy optimization suffers from vanishing gradients when all sampled rollouts yield identical outcomes. In single-turn reasoning, existing remedies mitigate this degenerate-group issue via prompt resampling (Yu et al., 2025), external scaffolding (Bamba et al., 2025; Zhang et al., 2026), virtual baselines (Nan et al., 2025), or entropy-guided surrogate advantages (Le et al., 2025). Moving to sequential decision-making(Cai et al., 2026), multi-turn search frameworks such as TSPO (Ma et al., 2026) and IGPO (Wang et al., 2025a) tackle credit assignment by designing heuristic step-level rewards or intermediate verifiers; however, they rely heavily on answer-dependent progress supervision or domain-specific scoring. While classic exploration methods (e.g., count-based bonuses or state-novelty measures like RIDE) encourage broader state visitation, they typically introduce non-stationary intrinsic rewards that risk distorting agent policies. However, these paradigms primarily cater to single-turn tasks or rely on task-specific heuristics, leaving the fundamental challenge of all-failure trajectories unresolved: in multi-step agent environments, extracting robust learning signals from purely negative trajectories is intrinsically difficult due to compound credit assignment. To our knowledge, CovGPO is the first framework tailored to resolve the all-failed regime in multi-turn agentic RL. Crucially, rather than enforcing answer-dependent intermediate supervision or altering the environment distribution, CovGPO quantifies relative credit via task-agnostic *observation coverage* directly recycled from existing on-policy interactions, serving as a plug-and-play auxiliary signal that seamlessly integrates into diverse comparison group structures (e.g., GRPO, GiGPO, and HGPO).

Auxiliary progress and exploration signals for agentic credit assignment. Sparse terminal rewards have motivated dense supervision from learned progress models, interaction structure, and intrinsic exploration. SPA-RL (Wang et al., 2025c) decomposes terminal outcomes into stepwise progress rewards, while iStar (Liu et al., 2026) and AgentPRM (Xi et al., 2026) learn process reward models for intermediate agent decisions. IGPO (Wang et al., 2025a) measures progress through information gain. Classical approaches such as RIDE (Raileanu & Rocktäschel, 2020) instead reward actions that induce substantial state changes. Unlike these persistent or model-based auxiliary signals, CovGPO uses model-free first-visit coverage only when outcome rewards provide no within-group ordering.

B PROOFS AND WORKED EXAMPLE

This section provides the rigorous theoretical foundations, complete proofs, and illustrative examples for Coverage-conditioned Group Policy Optimization (CovGPO). We formalize the theoretical properties of the coverage metric, establish gradient non-interference guarantees, analyze norm bounds and conditional expectations, and present a concrete step-by-step example on an all-failed rollout group.

Section Roadmap. To facilitate structured navigation, this section is organized into three major theoretical parts:

- **Theoretical Foundations of Coverage (Appendix B.1–B.4):** We prove exact reward degeneracy on zero-score rollout groups (Appendix B.1), formalize coverage exactness and necessity thresholds (Appendix B.2), establish how coverage deficit lower-bounds remaining work (Appendix B.3), and prove property ordering under coverage-monotone continuations (Appendix B.4).
- **Algorithmic Properties and Non-Interference (Appendix B.5–B.8):** We establish exact non-interference preserving reward-relative gradients on informative groups (Appendix B.5), prove the score-function identity under groupwise normalization (Appendix B.6), demonstrate frontier update boundaries via a counterexample (Appendix B.7), and derive static norm bounds for the auxiliary fallback gradient (Appendix B.8).
- **Disengagement, Gating Rationale, and Instantiation (Appendix B.9–B.11):** We derive rollout trigger probability bounds (Appendix B.9), justify the conditional fallback design against reward–progress conflicts (Appendix B.10), and walk through a detailed, concrete worked example on a representative all-failed group (Appendix B.11).

Common setup. Fix a task x in a finite-horizon environment. The transition and observation mechanisms, observation equality rule, reward $R(\tau) \geq 0$, completion indicator $S_x(\tau) \in \{0, 1\}$, and coverage statistic $P(\tau)$ have no explicit dependence on θ; their distributions depend on the policy. Write $\pi_\theta(\tau \mid x)$ for the induced trajectory law, suppressing x below. For score-function calculations, assume a differentiable law with locally common support and integrability permitting differentiation under the integral. Its score is $s_\theta(\tau) = \nabla_\theta \log \pi_\theta(\tau)$, and $s_i = s_\theta(\tau_i)$ for sampled trajectories. Independence is assumed only for the rollout sampling statements that explicitly require it. All empirical standard deviations use the population convention in Eq. equation 30. Unless specified otherwise, $\epsilon > 0$, $\tau_P > 0$, and $\lambda > 0$. Algorithm and loss comparisons hold sampled data, parameters, component membership, loss weights, reference policy, and regularization fixed, with advantages and gate statistics treated as constants during policy differentiation.

B.1 PROOF OF PROPOSITION 1

Setup. Recall $\mathcal{C}_H$ from Section 3.1: a^+ advances from s_j to s_{j+1}, a^- stays at s_j, s_H is absorbing, and $T = H$. A rollout succeeds exactly when all H actions are a^+. The fixed policy chooses a^+ with probability $\varepsilon \in (0, 1)$ at every nonterminal state.

Proof. Step 1: exact degeneracy. If every rollout has reward zero, then $\bar{R}_{G_x} = \sigma_R(G_x) = 0$ and Eq. equation 1 gives

$$A_i^R = \frac{0 - 0}{0 + \epsilon} = 0. \tag{8}$$

Every discounted step return is also zero, so centering within an anchor-state or history group gives zero step advantages. For any likelihood ratio r and clipping threshold ϵ_c, the associated surrogate term satisfies

$$\min\{rA_i^R, \mathrm{clip}(r, 1 - \epsilon_c, 1 + \epsilon_c)A_i^R\} = 0. \tag{9}$$

Thus these zero-advantage sample terms contribute zero reward-relative gradient.

Step 2: fixed-policy sampling. A rollout succeeds with probability ε^H. For independent groups of K independent rollouts from this fixed policy, the number N_{first} of groups up to and including the

first group containing a success is geometric with parameter $q = 1 - (1 - \varepsilon^H)^K$. The union bound gives

$$q \le K\varepsilon^H, \tag{10}$$

$$\mathbb{E}[N_{\text{first}}] = \frac{1}{q} = \frac{1}{1-(1-\varepsilon^H)^K} \ge \frac{1}{K\varepsilon^H}. \tag{11}$$

□

The waiting-time statement concerns repeated sampling from the specified fixed policy; the degeneracy statement concerns the zero-advantage surrogate terms on the collected data.

B.2 Coverage Exactness

Lemma 1 (Coverage identity and fixed-length necessity)**.** *Fix a task x and a length $T \ge 1$, and let $\mathcal{T}_{x,T}$ be its length-T trajectories. For $\tau \in \mathcal{T}_{x,T}$ with observations $(o_1, \dots, o_{T+1})$, including the observation returned by the last transition, let $D(\tau) = |\{o_1, \dots, o_{T+1}\}|$. Then*

$$P(\tau) = \frac{D(\tau) - 1}{T}. \tag{12}$$

Consequently, at equal length P orders trajectories exactly as D does, and no-op repetitions and cyclic revisits do not increment the coverage count. This identity requires no successful trajectory. For the necessity statement, assume additionally that $\{\tau \in \mathcal{T}_{x,T} : S_x(\tau) = 1\}$ is nonempty and define $m_{x,T} = \min\{D(\tau) : \tau \in \mathcal{T}_{x,T},\, S_x(\tau) = 1\} \ge 1$. Then

$$P(\tau) < \frac{m_{x,T} - 1}{T} \quad \Longrightarrow \quad S_x(\tau) = 0. \tag{13}$$

Proof. Setup. Fix the task and length domain of the Lemma and a trajectory $\tau \in \mathcal{T}_{x,T}$ with observation sequence $(o_1, \dots, o_{T+1})$, including the observation returned by the last transition. Let $\mathcal{O}(\tau) = \{o_1, \dots, o_{T+1}\}$ denote the set of distinct observations, so that $D(\tau) = |\mathcal{O}(\tau)|$, and let $\mathcal{V}_t = \{o_1, \dots, o_t\}$ be the prefix set. Recall from Eq. equation 5 that $\delta_t = \mathbf{1}[o_{t+1} \notin \mathcal{V}_t]$ and $P(\tau) = \frac{1}{T}\sum_{t=1}^{T} \delta_t$. For each $o \in \mathcal{O}(\tau)$, define its first-occurrence index

$$\rho(o) \;=\; \min\{s \in \{1, \dots, T+1\} : o_s = o\}. \tag{14}$$

Part (i). By definition of $\mathcal{V}_t$,

$$\delta_t = 1 \quad \Longleftrightarrow \quad o_{t+1} \notin \{o_1, \dots, o_t\} \quad \Longleftrightarrow \quad \rho(o_{t+1}) = t + 1. \tag{15}$$

Consider the map $\Phi : t \mapsto o_{t+1}$ restricted to the index set $\mathcal{T}_1 = \{t \in \{1, \dots, T\} : \delta_t = 1\}$. By Eq. equation 15, Φ maps $\mathcal{T}_1$ into $\{o \in \mathcal{O}(\tau) : \rho(o) \ge 2\}$; it is injective because two indices $t \ne t'$ in $\mathcal{T}_1$ satisfy $\rho(o_{t+1}) = t + 1 \ne t' + 1 = \rho(o_{t'+1})$, and it is surjective because every o with $\rho(o) = s \ge 2$ is the image of $t = s - 1 \in \mathcal{T}_1$. Hence Φ is a bijection, and since $\rho(o_1) = 1$ accounts for exactly one element of $\mathcal{O}(\tau)$,

$$\sum_{t=1}^{T} \delta_t \;=\; |\mathcal{T}_1| \;=\; \big|\{o \in \mathcal{O}(\tau) : \rho(o) \ge 2\}\big| \;=\; D(\tau) - 1, \qquad \text{i.e.,} \qquad P(\tau) = \frac{D(\tau) - 1}{T}. \tag{16}$$

A no-op repetition or cyclic revisit has $\delta_t = 0$ and does not increment the coverage count. Inserting a revisit into an otherwise unchanged observation sequence preserves D and increases the denominator, so the coverage rate weakly decreases.

Part (ii). Let $\tau \in \mathcal{T}_{x,T}$ be any trajectory with $S_x(\tau) = 1$. By the definition of the coverage requirement $m_{x,T} = \min\{D(\tau') : \tau' \in \mathcal{T}_{x,T},\, S_x(\tau') = 1\}$ in Lemma 1, $m_{x,T}$ is the minimum of D over successful length-T trajectories, hence $D(\tau) \ge m_{x,T}$. Combining with Part (i),

$$P(\tau) \;=\; \frac{D(\tau) - 1}{T} \;\ge\; \frac{m_{x,T} - 1}{T}, \tag{17}$$

which establishes the containment

$$\{\tau \in \mathcal{T}_{x,T} : S_x(\tau) = 1\} \subseteq \left\{\tau \in \mathcal{T}_{x,T} : P(\tau) \geq \tfrac{m_{x,T}-1}{T}\right\}, \tag{18}$$

and, taking complements within $\mathcal{T}_{x,T}$, the equivalent contrapositive form

$$P(\tau) < \frac{m_{x,T}-1}{T} \implies S_x(\tau) = 0. \tag{19}$$

□

Length and completion. The necessity threshold is specific to the task, observation representation, and length domain. The containment may be strict, since sufficient coverage need not complete the task. With unequal lengths, the rate P need not order raw coverage counts.

B.3 The Coverage Deficit Bounds the Remaining Work

Lemma 2 (Coverage deficit and successful continuation)**.** *Fix a total transition budget B and let $\mathcal{S}_{x,\leq B}$ be the set of all successful task-x trajectories of length at most B, assumed nonempty. Define*

$$m_{x,\leq B} = \min_{\tau' \in \mathcal{S}_{x,\leq B}} D(\tau'), \qquad d_{x,B}(\tau) = \max\{m_{x,\leq B} - D(\tau), 0\}. \tag{20}$$

Let τ be a length-t prefix, $0 \leq t \leq B$, admitting a valid, reset-free successful continuation η of length $\ell \leq B - t$ from its current state and retained history. Then

$$\ell \geq d_{x,B}(\tau). \tag{21}$$

In particular, an irreversibly terminated failure is not an admissible prefix under this hypothesis.

Proof. Step 1: count newly visited observations. The concatenation $\tau \circ \eta$ retains all prefix observations. Each of the ℓ additional transitions contributes at most one new observation, hence

$$D(\tau \circ \eta) = D(\tau) + |\mathcal{O}(\tau \circ \eta) \setminus \mathcal{O}(\tau)| \tag{22}$$

$$\leq D(\tau) + \ell. \tag{23}$$

Step 2: apply the budget-specific minimum. Since the concatenation succeeds and has length $t + \ell \leq B$,

$$m_{x,\leq B} \leq D(\tau \circ \eta) \leq D(\tau) + \ell, \tag{24}$$

$$\ell \geq \max\{m_{x,\leq B} - D(\tau), 0\} = d_{x,B}(\tau). \tag{25}$$

□

For equal positive prefix length t, the identity $D = 1 + tP$ yields $d_{x,B} = \max\{m_{x,\leq B} - 1 - tP, 0\}$. Thus increasing P weakly decreases this lower bound; the bound itself does not order continuation values or actual remaining work.

B.4 Ordering under Coverage-Monotone Continuation

The following additional hypothesis isolates an ordering property; it is not assumed by the algorithm or implied by the deficit bound.

Assumption 1 (Coverage-monotone continuation)**.** *Fix a task x, a policy π_θ, and a finite continuation budget Δ. For unsuccessful prefixes that admit reset-free resumption within the specified total horizon, let $V_\Delta(\tau)$ be the probability of completing x within Δ further steps from the endpoint state and retained history of τ. For equal-length such prefixes, assume $D(\tau) > D(\tau') \Rightarrow V_\Delta(\tau) \geq V_\Delta(\tau')$.*

Proposition 3 (Conditional ordering and empirical covariance)**.** *Let a triggered rollout group consist of equal-length trajectories satisfying Assumption 1. Write $V_i = V_\Delta(\tau_i)$ and $A_i = c(P_i - \bar{P})$ with a common $c > 0$. Then $\sum_i A_i = 0$, $P_i > P_j \Rightarrow A_i > A_j$, and $V_i > V_j \Rightarrow A_i \geq A_j$. Moreover, the population-convention empirical covariance satisfies $\mathrm{Cov}_{G_x}(A, V) \geq 0$.*

Proof. Step 1: centering and ordering. With common length T,

$$\sum_i A_i = c\left(\sum_i P_i - K\bar{P}\right) = 0, \tag{26}$$

$$A_i - A_j = c(P_i - P_j) = \frac{c}{T}(D_i - D_j). \tag{27}$$

If $V_i > V_j$ and $D_i < D_j$, Assumption 1 would imply $V_j \geq V_i$, a contradiction. Thus $V_i > V_j$ implies $D_i \geq D_j$ and $A_i \geq A_j$.

Step 2: empirical covariance. For every pair, $(A_i - A_j)(V_i - V_j) \geq 0$: unequal coverage invokes the assumption, while tied coverage gives equal advantages. Expanding the pairwise sum gives

$$\mathrm{Cov}_{G_x}(A, V) = \frac{1}{K}\sum_i A_i V_i - \bar{A}\bar{V} \tag{28}$$

$$= \frac{1}{2K^2}\sum_{i,j}(A_i - A_j)(V_i - V_j) \geq 0. \tag{29}$$

□

This is a fixed-group ordering result under Assumption 1, separate from the deficit Lemma. Policy-gradient improvement additionally depends on trajectory scores and the policy loss.

B.5 Proof of Proposition 2

Setup. Let $g^{\mathrm{CovGPO}}(G_x) = K^{-1}\sum_i A_i^{\mathrm{CovGPO}} s_i$ and $g^{\mathrm{base}}(G_x) = K^{-1}\sum_i A_i^R s_i$ denote the unweighted trajectory-score contributions. We use the population convention

$$\sigma_R^2(G_x) = \frac{1}{K}\sum_{i=1}^{K}(R_i - \bar{R}_{G_x})^2, \qquad \sigma_P^2(G_x) = \frac{1}{K}\sum_{i=1}^{K}(P_i - \bar{P}_{G_x})^2. \tag{30}$$

Proof. Step 1: rollout support. For the unweighted trajectory-score algorithms just defined, at the fixed task and policy,

$$\mathbb{E}[g^{\mathrm{CovGPO}}] - \mathbb{E}[g^{\mathrm{base}}] = \mathbb{E}_{G_x}\big[g^{\mathrm{CovGPO}}(G_x) - g^{\mathrm{base}}(G_x)\big]. \tag{31}$$

Partition the groups according to the two branches of Eq. equation 6:

$$\mathcal{G}_{\mathrm{fb}} = \{G_x : R_i = 0\ \forall i,\ \sigma_P(G_x) \geq \tau_P\}, \qquad \mathcal{G}_{\mathrm{base}} = \text{all other groups}. \tag{32}$$

Case $G_x \in \mathcal{G}_{\mathrm{base}}$. The exact-zero algorithm returns $A_i^{\mathrm{CovGPO}} = A_i^R$ for all i, hence the two group contributions are identical, including when reward dispersion is small but nonzero.

Case $G_x \in \mathcal{G}_{\mathrm{fb}}$. The gate supplies $R_1 = \cdots = R_K = 0$ directly, hence $\bar{R}_{G_x} = \sigma_R(G_x) = 0$ whatever the sign convention of the reward. Consequently,

$$A_i^R = \frac{R_i - \bar{R}_{G_x}}{\sigma_R(G_x) + \epsilon} = \frac{0 - 0}{0 + \epsilon} = 0 \qquad \text{for all } i, \qquad \text{hence} \qquad g^{\mathrm{base}}(G_x) = 0, \tag{33}$$

which proves the second statement. Combining the two cases, the integrand of Eq. equation 31 can be non-zero only on $\mathcal{G}_{\mathrm{fb}}$, where $g^{\mathrm{base}}(G_x) = 0$ identically. Thus the difference inside this expectation is supported only on groups whose reward-based contribution is identically zero.

Step 2: support after component aggregation. Index comparison components by c and let $\mathcal{E}$ be those eligible for exact-zero replacement. Their base scores are identically zero, so their centered base advantages vanish. Every component outside $\mathcal{E}$ is unchanged. For a loss sample n, let $\mathcal{C}(n)$ include all components on which its aggregated advantage or loss depends, and write its loss as $L_n(\theta; (A_c)_{c\in\mathcal{C}(n)})$. This map may include nonlinear aggregation and clipping. Define

$$\mathcal{N}_{\mathcal{E}} = \{n : \mathcal{C}(n) \cap \mathcal{E} \neq \emptyset\}. \tag{34}$$

For $n \notin \mathcal{N}_{\mathcal{E}}$ all inputs are identical, hence

$$L_n^{\mathrm{CovGPO}}(\theta) = L_n^{\mathrm{base}}(\theta), \tag{35}$$

$$\nabla_\theta L_n^{\mathrm{CovGPO}}(\theta) = \nabla_\theta L_n^{\mathrm{base}}(\theta) \tag{36}$$

where differentiable, with the same derivative convention at clipping boundaries. With fixed sample weights w_n and identical regularization,

$$L^{\mathrm{CovGPO}} - L^{\mathrm{base}} = \sum_{n \in \mathcal{N}_{\mathcal{E}}} w_n \big(L_n^{\mathrm{CovGPO}} - L_n^{\mathrm{base}}\big). \tag{37}$$

All potentially affected sample terms therefore depend on at least one eligible component. An affected sample may also depend on nonzero, unmodified components; nonlinear clipping does not decompose its gradient into a sum of component gradients. □

Lemma 3 (Score-gap equivalence of numerical and exact gates)**.** *Fix the comparison groups and their base scores y_i, with base advantages $(y_i - \bar{y})/(\sigma_y + \epsilon)$. Let $a = \tau_R^{\mathrm{impl}} > 0$ and assume*

$$|y_i| \in \{0\} \cup [a, \infty) \qquad \textit{for every comparison score}. \tag{38}$$

Compare the numerical wrapper to the exact-zero rule on the same groups and progress statistics, using the same effective scale $\lambda = \lambda_{\mathrm{eff}} \geq 0$, dispersion threshold $\tau_P = \tau_P^{\mathrm{impl}}$, and stability constant ϵ. Their eligibility gates, coverage triggers, and output advantages agree on every group, including the low-dispersion branch.

Proof. Step 1: eligibility. The magnitude gap and strict numerical threshold give

$$\max_i |y_i| < a \quad \Longleftrightarrow \quad y_i = 0 \text{ for all } i, \tag{39}$$

for scores of either sign. The identical progress statistics and cutoffs give identical coverage triggers. If an effective scale is computed from the eligible-group fraction, that fraction is also identical under the two eligibility tests.

Step 2: outputs. Ineligible groups retain the same base advantages. Eligible groups have $\bar{y} = \sigma_y = 0$, hence zero base advantages. Below the dispersion cutoff, the exact rule returns these zeros and the numerical rule explicitly returns zero. Otherwise, both return

$$A_i = \lambda_{\mathrm{eff}} \frac{P_i - \bar{P}}{\sigma_P + \epsilon}. \tag{40}$$

□

Corollary 1 (Score-gap sufficiency for gate equivalence)**.** *Write $a = \tau_R^{\mathrm{impl}} > 0$ and suppose that on the batch at hand every non-zero comparison score has magnitude at least a. Then $\max_{i \in G} |y_i| < a$ holds if and only if $y_i = 0$ for all $i \in G$, for every comparison group, so the numerical wrapper inherits the component-level conclusion of Proposition 2. For a binary terminal reward with $y_{i,t} = \gamma^{T_i - t} R_i$, $0 < \gamma \leq 1$ and $1 \leq t \leq T_i \leq T_{\max}$, the condition $\gamma^{T_{\max}-1} \geq a$ is sufficient; at $a = 10^{-3}$ this reads $\gamma \geq 0.869$ for $T_{\max} = 50$ and $\gamma \geq 0.611$ for $T_{\max} = 15$. More generally, if every positive terminal reward is at least $r_{\min} > 0$, then $\gamma^{T_{\max}-1} r_{\min} \geq a$ is sufficient.*

Proof. If any score in G is positive, the assumed gap gives $\max_i |y_i| \geq a$ and the numerical gate rejects the group; conversely an all-zero group satisfies $\max_i |y_i| = 0 < a$. The two gates therefore select the same groups, and Lemma 3 applies with $\lambda = \lambda_{\mathrm{eff}}$. For the horizon-based instance, a positive binary terminal reward is $R_i = 1$, so any positive comparison score satisfies $y_{i,t} = \gamma^{T_i - t} \geq \gamma^{T_{\max}-1}$ because $T_i - t \leq T_{\max} - 1$ and $0 < \gamma \leq 1$; the same monotonicity gives $y_{i,t} \geq \gamma^{T_{\max}-1} r_{\min}$ whenever every positive terminal reward is at least $r_{\min}$. Solving $\gamma^{T_{\max}-1} \geq a$ for $T_{\max} > 1$ yields $\gamma \geq a^{1/(T_{\max}-1)}$, i.e. $\gamma \geq 10^{-3/49} = 0.8686$ at $T_{\max} = 50$ and $\gamma \geq 10^{-3/14} = 0.6106$ at $T_{\max} = 15$; the thresholds above are rounded upward so that they remain sufficient. □

What the corollary does and does not establish. The corollary is a statement about a score gap, not about a particular run. Applying it requires reading γ, $T_{\max}$ and the comparison-score definition from the configuration actually used, rather than from library defaults. Under the stated discount condition, the positive-score lower bound is at least the numerical threshold τ_R^{impl}, so the two gates coincide. For fractional WebShop rewards the condition additionally needs a confirmed global lower bound $r_{\min} > 0$ on positive terminal rewards satisfying $\gamma^{T_{\max}-1} r_{\min} \geq 10^{-3}$. A minimum positive reward observed in one set of logs certifies only that recorded data, and if positive rewards can be arbitrarily close to zero no admissible $r_{\min}$ exists. In that case the available check is to count groups accepted by the numerical gate while containing a positive score; such a count establishes consistency on the recorded runs, not a guarantee on future data.

Near-zero scores. Without the score-gap hypothesis, the numerical rule preserves base advantages on groups with $\max_i |y_i| \geq a$. Its additional modification region is $0 < \max_i |y_i| < a$, and the displaced contrast there need not be small. Group normalization divides by the within-group standard deviation, so the magnitude of the score gap cancels: for $K = 2$ and $y = (0, 5 \times 10^{-4})$ with $a = 10^{-3}$ and $\epsilon = 10^{-6}$, the base advantages are ∓ 0.996, close to the full scale of a two-member group, yet the group is numerically eligible and the wrapper replaces those advantages by coverage advantages, or by zero when coverage is homogeneous. The exact rule of Eq. equation 6 retains them. The sample-support argument above applies with numerical eligibility in place of $\mathcal{E}$, but eligible base components need not vanish, so the non-interference statement does not transfer to the wrapper on this region.

Scope of Non-Interference Guarantee. For the ideal exact-zero rule, non-triggered comparison components are unchanged, and loss samples that depend on no triggered component retain the same value and gradient. The deployed wrapper uses a near-zero gate ($\max_{i \in G} |y_i| < 10^{-3}$); transferring the exact non-interference guarantee requires the score-gap condition derived above.

Binary-outcome specialization. For raw binary rollout rewards and $K \geq 2$, the smallest nonzero population standard deviation is

$$\sigma_{R,\min} = \min_{1 \leq m \leq K-1} \frac{\sqrt{m(K-m)}}{K} = \frac{\sqrt{K-1}}{K}. \tag{41}$$

The numerical max-absolute-score test selects exactly the all-zero groups when its threshold lies in $(0, 1]$, because every positive raw binary reward is 1. This equivalence follows from the score gap, not from a standard-deviation test. It need not hold for discounted step returns or fractional WebShop scores. Uniform positive-reward groups remain outside either fallback gate despite having zero relative advantages.

B.6 The Unnormalized REINFORCE Identity and Groupwise Normalization

Lemma 4 (Unnormalized i.i.d. identity and groupwise normalization)**.** *Fix x and θ, let $F = \{R(\tau) = 0\}$ with $\Pr(F) > 0$, and assume differentiable trajectory probabilities, a parameter-independent reward event and coverage statistic, and conditions permitting differentiation under expectation. For $K \geq 2$ i.i.d. rollouts, define $F_K = F^K$, $s_i = \nabla_\theta \log \pi_\theta(\tau_i)$, and*

$$\widehat{g}(G_x) = \frac{1}{K} \sum_i (P_i - \bar{P}_{G_x}) s_i. \tag{42}$$

Before conditioning on coverage dispersion or normalizing by its random standard deviation,

$$\mathbb{E}[\widehat{g}(G_x) \mid F_K] = \frac{K-1}{K} \nabla_\theta \mathbb{E}_{\pi_\theta(\cdot|F)}[P]. \tag{43}$$

Separately, for a fixed group admitted by the ideal fallback, its unweighted, unclipped trajectory-score contribution is $g(G_x) = c(G_x)\widehat{g}(G_x)$ with $c(G_x) = \lambda/(\sigma_P(G_x) + \epsilon) > 0$.

Proof. Step 1: fixed-group factorization. For a group entering the fallback branch of Eq. equation 6, substitute $A_i^{\mathrm{CovGPO}} = \lambda(P_i - \bar{P})/(\sigma_P + \epsilon)$ to obtain

$$g(G_x) = \frac{1}{K}\sum_i A_i^{\mathrm{CovGPO}} s_i \tag{44}$$

$$= \frac{\lambda}{\sigma_P(G_x) + \epsilon}\left[\frac{1}{K}\sum_i (P_i - \bar{P}_{G_x}) s_i\right] = c(G_x)\widehat{g}(G_x). \tag{45}$$

Step 2: conditional trajectory score. Define the failure normalizer and conditional law by

$$Z_F(\theta) = \Pr_{\pi_\theta}(F) = \int_F \pi_\theta(\tau)\, d\tau > 0, \tag{46}$$

$$\pi_\theta^F(\tau) = \frac{\mathbf{1}_F(\tau)\pi_\theta(\tau)}{Z_F(\theta)}, \qquad \mu_F(\theta) = \mathbb{E}_{\pi_\theta^F}[P]. \tag{47}$$

Here integrals also denote sums for discrete trajectory spaces, and $\mathbb{E}_F$ abbreviates expectation under π_θ^F. Differentiation under the integral gives

$$\nabla_\theta \log Z_F(\theta) = \frac{1}{Z_F(\theta)}\int_F \pi_\theta(\tau) s_\theta(\tau)\, d\tau = \mathbb{E}_F[s_\theta], \tag{48}$$

$$\nabla_\theta \log \pi_\theta^F(\tau) = s_\theta(\tau) - \nabla_\theta \log Z_F(\theta), \qquad \tau \in F. \tag{49}$$

Since P has no explicit parameter dependence,

$$\nabla_\theta \mu_F(\theta) = \mathbb{E}_F\big[P\big(s_\theta - \nabla_\theta \log Z_F\big)\big] \tag{50}$$

$$= \mathbb{E}_F[P s_\theta] - \mathbb{E}_F[P]\mathbb{E}_F[s_\theta] = \mathrm{Cov}_{\pi_\theta^F}(P, s_\theta). \tag{51}$$

Step 3: the group-mean factor. Independence of the original rollouts implies

$$\pi_\theta(\tau_1, \ldots, \tau_K \mid F_K) = \prod_{i=1}^K \pi_\theta^F(\tau_i). \tag{52}$$

Thus diagonal and off-diagonal terms in the double sum have different expectations:

$$\mathbb{E}[\widehat{g} \mid F_K] = \frac{1}{K}\sum_i \mathbb{E}_F[P_i s_i] - \frac{1}{K^2}\sum_{i,j}\mathbb{E}[P_j s_i \mid F_K] \tag{53}$$

$$= \mathbb{E}_F[Ps] - \frac{1}{K^2}\left(\sum_{i=j}\mathbb{E}_F[Ps] + \sum_{i\neq j}\mathbb{E}_F[P]\mathbb{E}_F[s]\right) \tag{54}$$

$$= \mathbb{E}_F[Ps] - \frac{K}{K^2}\mathbb{E}_F[Ps] - \frac{K(K-1)}{K^2}\mathbb{E}_F[P]\mathbb{E}_F[s] \tag{55}$$

$$= \frac{K-1}{K}\,\mathrm{Cov}_{\pi_\theta^F}(P, s) = \frac{K-1}{K}\nabla_\theta \mu_F(\theta). \tag{56}$$

□

Normalized algorithm. Equation equation 56 applies to the auxiliary unnormalized statistic $\widehat{g}$, whereas the algorithm uses dispersion-gated, normalized advantages. For the ideal rule,

$$\mathbb{E}[g_{\mathrm{fb}} \mid F_K] = \mathbb{E}\left[\mathbf{1}[\sigma_P \geq \tau_P]\frac{\lambda}{\sigma_P + \epsilon}\widehat{g}\,\middle|\, F_K\right]. \tag{57}$$

The random multiplier remains inside the expectation. The numerical wrapper additionally uses batch-dependent λ_{eff} and numerical eligibility. Positive scaling preserves a fixed admitted group's unweighted score-sum direction, but neither random scaling nor selection can be removed from an expectation by Eq. equation 56. Length-weighted or clipped losses and dependent step-comparison groups require their own algorithms.

B.7 A Counterexample to Fallback-Only Escape Guarantees

Proposition 4 (Negative frontier update among failures). *On $\mathcal{C}_2$, let $p_j = \pi_\theta(a^+ \mid s_j) = \varsigma(z_j)$ for independent logistic parameters z_0, z_1, and set $p_0 = p_1 = 1/2$. Conditional on failure $F = \{R = 0\}$, the coverage–score covariance for the frontier logit z_1 is $-1/36$. Thus even the unnormalized expected fallback update need not increase every forward-action logit.*

Proof. Step 1: conditional covariance. With observations identifying chain states, a length-H trajectory attaining depth $d(\tau)$ has

$$P(\tau) = \frac{D(\tau) - 1}{H} = \frac{d(\tau)}{H}. \tag{58}$$

At $H = 2$ and $p_0 = p_1 = 1/2$, the four action sequences each have probability $1/4$, and only $++$ succeeds. The failed sequences $--, -+, +-$ are therefore equiprobable conditional on F. In this order,

$$(P(--), P(-+), P(+-)) = (0, \tfrac{1}{2}, \tfrac{1}{2}), \tag{59}$$

$$(\partial_{z_1} \log \pi(--), \partial_{z_1} \log \pi(-+), \partial_{z_1} \log \pi(+-)) = (0, 0, -\tfrac{1}{2}). \tag{60}$$

The first two trajectories take no action at s_1; the last chooses a^- there, whose logistic score is $-p_1 = -1/2$. Consequently

$$\mathrm{Cov}_{\pi_\theta(\cdot|F)}\left(P, \partial_{z_1} \log \pi_\theta\right) = -\frac{1}{12} - \left(\frac{1}{3}\right)\left(-\frac{1}{6}\right) = -\frac{1}{36}. \tag{61}$$

By Lemma 4, the unnormalized i.i.d. group algorithm has expected z_1 component $-(K-1)/(36K)$ conditional on all rewards being zero, for $K > 1$.

Step 2: finite-group covariance. For any all-failed group on this chain, let u be the fraction of members equal to $--$ and v the fraction equal to $+-$, and write $s_{i,1} = \partial_{z_1} \log \pi_\theta(\tau_i)$. The remaining fraction $1 - u - v$ consists of $-+$. Hence

$$\bar{P} = \frac{1-u}{2}, \qquad \bar{s}_1 = -\frac{v}{2}, \quad \frac{1}{K}\sum_i P_i s_{i,1} = -\frac{v}{4}, \tag{62}$$

$$\frac{1}{K}\sum_i (P_i - \bar{P}) s_{i,1} = -\frac{v}{4} - \frac{1-u}{2}\left(-\frac{v}{2}\right) = -\frac{uv}{4} \le 0. \tag{63}$$

A group containing one copy of each failed sequence has $u = v = 1/3$, covariance $-1/36$, and $\sigma_P = \sqrt{1/18}$. For any admitted group, positive normalization multiplies the last expression by $\lambda/(\sigma_P + \epsilon) > 0$; a rejected group has zero fallback contribution. Neither operation makes its frontier component positive. □

Implication. Conditioning on failure removes $++$, which credits the final forward action, while a higher-coverage failure can credit staying at the frontier. Thus positive coverage dispersion does not supply the positive frontier drift needed for a fallback-only polynomial escape argument. Reward-based updates on successful groups remain separate from this failed-group calculation.

B.8 A Static Norm Bound for the Gated Fallback

Proposition 5 (Groupwise auxiliary norm bound). *For a fixed rollout group, define $I_{\mathrm{fb}}(G_x) = \mathbf{1}[R_i = 0\ \forall i,\ \sigma_P(G_x) \ge \tau_P]$ and the auxiliary score-function contribution*

$$g_{\mathrm{fb}}(G_x) = I_{\mathrm{fb}}(G_x) \frac{\lambda}{\sigma_P(G_x) + \epsilon} \frac{1}{K} \sum_i (P_i - \bar{P}_{G_x}) \nabla_\theta \log \pi_\theta(\tau_i). \tag{64}$$

Writing $M(G_x) = \max_i \|\nabla_\theta \log \pi_\theta(\tau_i)\|$, one has

$$\|g_{\mathrm{fb}}(G_x)\| \le I_{\mathrm{fb}}(G_x) \frac{\lambda \sigma_P(G_x)}{\sigma_P(G_x) + \epsilon} M(G_x) \le I_{\mathrm{fb}}(G_x) \lambda M(G_x). \tag{65}$$

In particular, $g_{\mathrm{fb}} = 0$ outside the fallback gate, including when $\sigma_P < \tau_P$. This bounds the auxiliary contribution, not the full base-plus-fallback gradient.

Proof. Step 1: triangle inequality. Writing $s_i = \nabla_\theta \log \pi_\theta(\tau_i)$,

$$\|g_{\mathrm{fb}}(G_x)\| \le I_{\mathrm{fb}} \frac{\lambda}{\sigma_P + \epsilon} \frac{1}{K} \sum_i |P_i - \bar{P}| \, \|s_i\| \tag{66}$$

$$\le I_{\mathrm{fb}} \frac{\lambda M(G_x)}{\sigma_P + \epsilon} \frac{1}{K} \sum_i |P_i - \bar{P}|. \tag{67}$$

Step 2: Cauchy–Schwarz. The population convention yields

$$\left(\frac{1}{K} \sum_i |P_i - \bar{P}| \right)^2 \le \left(\frac{1}{K} \sum_i 1^2 \right) \left(\frac{1}{K} \sum_i (P_i - \bar{P})^2 \right) = \sigma_P^2, \tag{68}$$

$$\|g_{\mathrm{fb}}(G_x)\| \le I_{\mathrm{fb}} \frac{\lambda \sigma_P}{\sigma_P + \epsilon} M(G_x) \le I_{\mathrm{fb}} \lambda M(G_x). \tag{69}$$

The indicator makes the contribution zero outside the gate. □

Uniform bounds. If the additional score-norm condition $\sup_n \max_{i \in G_n} \|s_{\theta_n}(\tau_i)\| \le M_*$ holds along training, then the fixed-λ auxiliary contributions satisfy $\sup_n \|g_{\mathrm{fb}}(G_n)\| \le \lambda M_*$. Bounded coverage alone does not supply this condition, imply shrinking coverage variance, or exclude coverage-seeking attractors.

Numerical coverage term. Replacing I_{fb} by the numerical eligibility-and-dispersion indicator and λ by $\lambda_{\mathrm{eff}} \le \lambda_{\mathrm{aux}}$ gives the analogous bound $\|g_{\mathrm{cov}}^{\mathrm{impl}}\| \le \lambda_{\mathrm{aux}} M(G_x)$. This bounds the coverage score-function term; the difference from the base also includes any removed near-zero base advantages, and the optimizer additionally applies loss weighting and clipping.

B.9 Disengagement Bounds: Two Conditions

Remark 1 (Exact-zero and numerical rollout trigger probabilities)**.** *Fix a task x and policy π_θ, and let $p_+(x) = \Pr_{\pi_\theta}(R > 0 \mid x)$. For K i.i.d. rollouts, define $F_K = \{R_1 = \cdots = R_K = 0\}$ and $\mathcal{D} = \{\sigma_P(G_x) \ge \tau_P\}$. If $p_+(x) < 1$, then*

$$\Pr(\mathcal{E}_{\mathrm{trig}}(x)) = \Pr(F_K \cap \mathcal{D}) = (1 - p_+(x))^K \Pr(\mathcal{D} \mid F_K) \tag{70}$$

$$\le (1 - p_+(x))^K \le e^{-K p_+(x)}. \tag{71}$$

If $p_+(x) = 1$, the trigger probability is zero. For the numerical rollout gate, let $a = \tau_R^{\mathrm{impl}}$, $q_a(x) = \Pr(|R| < a \mid x)$, $E_{a,K} = \{|R_i| < a \; \forall i\}$, and $\mathcal{D}_{\mathrm{impl}} = \{\sigma_P \ge \tau_P^{\mathrm{impl}}\}$. When $q_a(x) > 0$,

$$\Pr(\mathcal{E}_{\mathrm{trig}}^{\mathrm{impl}}(x)) = q_a(x)^K \Pr(\mathcal{D}_{\mathrm{impl}} \mid E_{a,K}) \le q_a(x)^K; \tag{72}$$

when $q_a(x) = 0$, this probability is zero.

Proof. Step 1: reward eligibility. Independent rollout sampling and $\Pr(R_i = 0 \mid x) \le 1 - p_+(x)$, with equality when rewards are non-negative, imply

$$\Pr(F_K) = \prod_{i=1}^K \Pr(R_i = 0 \mid x) \le (1 - p_+(x))^K, \tag{73}$$

$$\Pr(E_{a,K}) = \prod_{i=1}^K \Pr(|R_i| < a \mid x) = q_a(x)^K. \tag{74}$$

Step 2: intersect with dispersion. For any event E of positive probability,

$$\Pr(E \cap D) = \Pr(E) \Pr(D \mid E) \le \Pr(E). \tag{75}$$

Apply this identity to $(F_K, \mathcal{D})$ and $(E_{a,K}, \mathcal{D}_{\mathrm{impl}})$, respectively, and use $1 - p \le e^{-p}$ for $p \in [0, 1]$. If an eligibility event has zero probability, so does its intersection with dispersion, without defining a conditional probability on that null event. □

The factorization uses independence across rollouts, not independence between reward eligibility and dispersion. The two conditions jointly determine triggering at a fixed policy; neither factor is assumed to evolve monotonically during training. Here p_+ measures positive reward, which equals success probability for binary outcomes but not generally for fractional rewards. Dependent state- or history-selected comparison groups do not satisfy the rollout product formula without an additional sampling assumption.

B.10 Why the Fallback Must Be Conditional: Reward–Progress Conflict

The conditional design in Eq. equation 6 is motivated by a specific mismatch between reward and progress rankings that can arise in mixed-outcome groups. In such a group $G_x = \{\tau_i\}_{i=1}^K$, the reward-based ranking is monotone in $R_i \in \{0, 1\}$: any successful trajectory ranks above any failed one. The progress score $P_i = (D_i - 1)/T_i$, by contrast, is a length-normalized coverage rate and is not monotone in R_i—a short, efficient success can score lower than a long, wandering failure.

Concrete instance. Consider $G_x = \{\tau_1, \tau_2\}$ with

- τ_1 (efficient success): R_1=1, T_1=5, D_1=3, hence P_1=0.4;
- τ_2 (wandering failure): R_2=0, T_2=10, D_2=7, hence P_2=0.6.

Ranking calculation. Define $A_i^P = (P_i - \bar{P})/(\sigma_P + \epsilon)$ and the always-on additive algorithm $A_i = A_i^R + \lambda A_i^P$. The reward ranking reverses in this example exactly when

$$\lambda > \lambda_* := \frac{0.1 + \epsilon}{0.2(0.5 + \epsilon)}. \tag{76}$$

Proof. The population means and standard deviations are $\bar{R} = \sigma_R = 0.5$ and $\bar{P} = 0.5$, $\sigma_P = 0.1$. Therefore

$$(A_1^R, A_2^R) = \frac{(0.5, -0.5)}{0.5 + \epsilon}, \qquad (A_1^P, A_2^P) = \frac{(-0.1, 0.1)}{0.1 + \epsilon}, \tag{77}$$

$$A_1 - A_2 = \frac{1}{0.5 + \epsilon} - \lambda \frac{0.2}{0.1 + \epsilon}. \tag{78}$$

Thus $A_1 > A_2$ for $\lambda < \lambda_*$, they tie at $\lambda = \lambda_*$, and $A_1 < A_2$ for $\lambda > \lambda_*$. Omitting ϵ gives $A = (1 - \lambda, -1 + \lambda)$ and $\lambda_* = 1$. □

This calculation concerns advantage rankings; changes in trajectory probabilities under shared parameters also depend on their score vectors and the loss. The exact-zero rule preserves the reward advantages in this mixed-outcome group, as does the numerical rule for these raw binary scores. Lemma 3 specifies when this agreement extends to other comparison scores. Table 2 gives the empirical comparison with always-on alternatives.

B.11 A Worked all-failed Group: Instantiating the Main-Text Results

We use a schematic ALFWorld-style goal, "*put a cool tomato on the countertop*," to illustrate Eq. equation 6 with $K = 4$ zero-reward trajectories of equal length $T_i = 10$:

- τ_1 (goal-directed): `go to fridge 1` → `open fridge 1` → `take tomato 1` → `cool tomato 1 with fridge 1` → navigates toward the countertop until the step budget expires. 8 of 10 steps produce a *previously unseen* observation.
- τ_2 (partial explorer): reaches and opens the fridge, then wanders; 5 novel observations, plus one revisit of an earlier room.
- τ_3 (cycler): shuttles back and forth between two rooms. The observation changes at 9 of 10 steps, but only 2 of those observations are novel—the rest revisit states already in $\mathcal{V}_{3,t}$.
- τ_4 (no-op): repeats `look`/`inventory`; the observation changes once in 10 steps.

Table 4: A worked $K = 4$ all-zero-reward group ($T_i = 10$; $\tau_P = 0.01$, $\lambda = 1$, ϵ omitted). The novelty gate counts first visits, whereas Obs-Change ranks the cycler first. These are score rankings, not certified task-value rankings.

Traj.	Behaviour	Obs.-changing steps	Novel steps	P_i	A_i^{CovGPO}	$A_i^{\text{Obs-Change}}$
τ_1	goal-directed	8	8	0.8	$+1.46$	$+0.65$
τ_2	partial explorer	6	5	0.5	$+0.37$	0.00
τ_3	cycler	9	2	0.2	-0.73	$\mathbf{+0.97}$
τ_4	no-op	1	1	0.1	-1.10	-1.62

Table 4 reports, for each trajectory, the number of observation-*changing* steps, the number of *novelty-gated* steps counted by Eq. equation 5, the resulting progress score P_i (Eq. equation 5), and the advantages assigned by CovGPO versus the ungated Obs-Change variant from the ablation in Section 4.

Branch selection and fallback advantages. With $R = (0, 0, 0, 0)$ and $P = (0.8, 0.5, 0.2, 0.1)$,

$$\bar{P} = 0.4, \qquad \sigma_P = \sqrt{\frac{0.4^2 + 0.1^2 + (-0.2)^2 + (-0.3)^2}{4}} = \sqrt{0.075} \approx 0.274. \tag{79}$$

The dispersion exceeds $\tau_P = 0.01$. For $\lambda = 1$ and omitted ϵ, centering and normalization give $(+1.46, +0.37, -0.73, -1.10)$. Lemma 1 also gives $D = (9, 6, 3, 2)$. These counts determine the coverage ranking; continuation-value ordering would additionally require Assumption 1. Deployed magnitudes use λ_{eff} and the stability constant instead of this illustrative unit scale.

Novelty versus observation change. Obs-Change uses scores $(0.8, 0.6, 0.9, 0.1)$, with mean 0.6 and population standard deviation $\sqrt{0.095}$. It therefore assigns $(+0.65, 0.00, +0.97, -1.62)$ and ranks the cycler first. First-visit coverage instead credits only 2 of the cycler's 9 observation changes.

Inactive branches. Replacing the rewards by $(1, 0, 0, 0)$ returns base advantages $(+1.73, -0.58, -0.58, -0.58)$ with ϵ omitted; both reward gates preserve this binary-score group. Alternatively, a zero-reward group with $P = (0.8, 0.8, 0.8, 0.8)$ has zero dispersion and receives zero under both rules. These instantiate the reward-eligibility and dispersion conditions in Remark 1.

C Implementation Details

This section provides complete implementation specifications, computational hardware details, training configurations, and environment setups required for empirical reproducibility. We detail how CovGPO seamlessly integrates into group-based RL algorithms as a plug-and-play advantage modification, present reference pseudo-code, and formalize our evaluation protocols.

Section Roadmap. The implementation details are organized as follows:

- **Infrastructure and Algorithmic Specifications (Appendix C.1–C.3):** We outline computing infrastructure and runtime overhead (Appendix C.1), provide reference algorithmic implementations for exact and deployed wrappers (Appendix C.2), and define adaptive per-update scaling boundaries (Appendix C.3).
- **Hyperparameters and Environment Protocols (Appendix C.4–C.7):** We detail overall model training hyperparameters (Appendix C.4), describe simulation setups for ALFWorld (Appendix C.5) and WebShop (Appendix C.6), and specify cross-environment progress metrics (Appendix C.7).
- **Evaluation and Reporting Protocols (Appendix C.8):** We formalize evaluation protocols, inference decoding parameters, seed aggregation rules, and evaluation settings.

Algorithm 1 Ideal exact-zero CovGPO advantages for one rollout group. Outside the fallback branch, return the unchanged base advantage.

Require: Trajectories $G_x = \{\tau_i\}_{i=1}^{K}$ with $T_i \geq 1$; non-negative rewards R_i; $\tau_P > 0$, $\lambda > 0$, $\epsilon > 0$

for $i = 1, \ldots, K$ **do**
  $\mathcal{V}_i \leftarrow \{o_{i,1}\}$; $C_i \leftarrow 0$
  for $t = 1, \ldots, T_i$ **do**
    if $o_{i,t+1} \notin \mathcal{V}_i$ **then**
      $C_i \leftarrow C_i + 1$
    end if
    $\mathcal{V}_i \leftarrow \mathcal{V}_i \cup \{o_{i,t+1}\}$
  end for
  $P_i \leftarrow C_i / T_i$
end for
Compute $\bar{R}, \sigma_R, \bar{P}, \sigma_P$ using Eq. equation 30
if $R_i = 0$ for every i **and** $\sigma_P \geq \tau_P$ **then**
  for $i = 1, \ldots, K$ **do**
    $A_i \leftarrow \lambda(P_i - \bar{P})/(\sigma_P + \epsilon)$
  end for
else
  for $i = 1, \ldots, K$ **do**
    $A_i \leftarrow (R_i - \bar{R})/(\sigma_R + \epsilon)$
  end for
end if
return $\{A_i\}_{i=1}^{K}$ and branch identifier

C.1 Computing Details

CovGPO is implemented as a drop-in replacement of the advantage computation in the group-based RL trainer; no change is made to rollout collection, the optimizer, or the model. For each rollout group, computing the trajectory progress scores $\{P_i\}$ requires maintaining a per-trajectory hash set of visited observation strings and performing expected $O(1)$ hash-set membership lookups per step after processing each observation key, executed on CPU during advantage estimation.

Each baseline–CovGPO pair is trained on 4 GPUs for the 1.5B configuration and 8 GPUs for the 7B configuration, with the same configuration within each matched pair. The reported progress-and-switch overhead is less than 0.5% of per-epoch wall-clock time. This is a reported timing summary, not a verified noise estimate or a hardware-independent bound; rollout generation and policy updates dominate the reported budget. The stage-level timing audit in Figure 5 was run separately on four GPUs and is therefore used to measure overhead, not to redefine the training hardware above. CovGPO adds no learned parameters or persistent model-state memory (no critic, reward model, progress model, or world model); its visited sets reside in host memory and its advantage tensor has the same shape as that of the base trainer.

C.2 Reference Advantage Computation

Algorithm 1 gives the ideal exact-zero rollout algorithm. Algorithm 2 separately records the numerical wrapper used in training. Observations are compared as the exact strings emitted by the environment: no lower-casing, whitespace normalization, tokenization, semantic encoder, or environment-specific equivalence rule is applied. The initial observation is inserted into the visited set before the first action. A terminal observation returned by the final environment step is processed exactly like every other next observation. Trajectory length T_i is the number of executed environment actions, including the terminating action when termination occurs before the maximum horizon.

C.3 Deployed Per-Update Scaling and Conformance Boundary

Algorithm 1 is the equation-level algorithm analyzed in Section B. The deployed wrapper differs in both numerical eligibility and the low-dispersion branch, and computes a common non-negative

Algorithm 2 Deployed CovGPO wrapper for one policy update. The base method determines the group keys: prompt groups for GRPO, anchor-state groups for GiGPO, and history groups for HGPO.

Require: Groups $\mathcal{B}$, base scores $\{y_i\}$, progress $\{p_i\}$, τ_R^{impl}, τ_P^{impl}, λ_{aux}, ϵ

$n_{\mathrm{fail}} \leftarrow 0$
for each $G \in \mathcal{B}$ **do**
  $f_G \leftarrow \mathbf{1}\left[\max_{i \in G} |y_i| < \tau_R^{\mathrm{impl}}\right]$
  $n_{\mathrm{fail}} \leftarrow n_{\mathrm{fail}} + f_G$
end for
$q_{\mathrm{fail}} \leftarrow n_{\mathrm{fail}} / \max(|\mathcal{B}|, 1)$
$\lambda_{\mathrm{eff}} \leftarrow \lambda_{\mathrm{aux}} q_{\mathrm{fail}}$
for each $G \in \mathcal{B}$ **do**
  Compute unchanged base advantages A_i^{base} for $i \in G$
  if $f_G = 1$ **then**
    $\bar{p}_G \leftarrow |G|^{-1} \sum_{i \in G} p_i$
    $s_{p,G} \leftarrow \mathrm{Std}_{\mathrm{pop}}(\{p_i : i \in G\})$
    if $s_{p,G} \geq \tau_P^{\mathrm{impl}}$ **then**
      for each $i \in G$ **do**
        $A_i \leftarrow \lambda_{\mathrm{eff}}(p_i - \bar{p}_G)/(s_{p,G} + \epsilon)$
      end for
    else
      for each $i \in G$ **do**
        $A_i \leftarrow 0$
      end for
    end if
  else
    for each $i \in G$ **do**
      $A_i \leftarrow A_i^{\mathrm{base}}$
    end for
  end if
end for
return $\{A_i\}$ and diagnostics $(q_{\mathrm{fail}}, \lambda_{\mathrm{eff}}, n_{\mathrm{fail}})$

scale once per policy update. Let $\mathcal{B}$ be the comparison groups produced by the selected base method, and define

$$q_{\mathrm{fail}}(\mathcal{B}) = \frac{1}{|\mathcal{B}|} \sum_{G \in \mathcal{B}} \mathbf{1}\left[\max_{i \in G} |y_i| < \tau_R^{\mathrm{impl}}\right], \qquad \lambda_{\mathrm{eff}} = \lambda_{\mathrm{aux}} q_{\mathrm{fail}}(\mathcal{B}), \tag{80}$$

where $\mathcal{B}$ is nonempty, y_i is the reward or return supplied by the base algorithm, and $\tau_R^{\mathrm{impl}} = 10^{-3}$. The fraction q_{fail} is measured before the coverage-dispersion test and denotes numerical eligibility rather than a success-based failure fraction. The released runs set `lambda_fixed=false`; consequently, λ_{eff} changes with the current update even though its configured ceiling λ_{aux} is fixed. For $\lambda_{\mathrm{aux}} > 0$, any update containing an eligible group has $\lambda_{\mathrm{eff}} > 0$. Its common scaling preserves each admitted centered coverage vector up to magnitude, without implying an unbiased objective gradient. If $q_{\mathrm{fail}} = 0$, no group is eligible and the fallback is exactly inactive.

Batch-level dependence of the effective scale. Equation 6 treats λ as a constant, whereas Eq. equation 80 recomputes it from the current batch. We state what this does and does not change. It does not affect Proposition 2: the scale multiplies only the fallback branch and the branch selection does not depend on it, so non-interference holds for every $\lambda_{\mathrm{eff}} \geq 0$, including a batch-dependent one. It does not affect the norm bound of Proposition 5 either, because $\lambda_{\mathrm{eff}} \leq \lambda_{\mathrm{aux}}$ and the bound may be stated with the ceiling in place of λ. Within a single update, λ_{eff} is one non-negative number shared by every triggered group, hence a common rescaling that preserves both the relative ordering of those groups and the direction of each admitted centered coverage vector.

What it does introduce is a dependence on batch composition. The advantage assigned to a triggered group becomes a function of the batch in which that group appears rather than of the group alone:

two identical all-failed groups occurring in different updates receive different magnitudes, and the assignment is no longer the fixed-λ map of Eq. 6. Three consequences follow. First, λ_{aux} is a ceiling and not the operative strength, so the sweep in Appendix E.7 varies the ceiling; a setting that underperforms there does not isolate the effect of a larger realized auxiliary gradient. Second, q_{fail} is an empirical fraction over the groups of one update, so its sampling noise, and therefore the variability of the effective scale, depends on how many comparison groups an update contains; this differs between trajectory-level and step-level base methods. Third, no expectation argument may treat λ_{eff} as a constant factor, since it is correlated with the batch that supplies the triggered groups; this is the random multiplier already flagged after Eq. equation 56.

Finally, because q_{fail} declines as rewarded rollouts reappear, the auxiliary term is in effect annealed by trigger frequency rather than by a designed schedule. We record this as a property of the existing implementation. It is not a convergence argument, and Appendix E.8.4 notes separately that a declining branch frequency does not by itself verify a declining multiplier. If $q_{\text{fail}} = 0$ the fallback is exactly inactive and the update coincides with the base method.

Conformance boundary. The ideal gate tests exact zeros; the deployed gate tests $\max_{i \in G} |y_i| < 10^{-3}$. Non-negative scores do not make them equivalent. An eligible group with small positive-score contrast receives coverage advantages if dispersion passes, and a hard zero otherwise. Thus the deployed wrapper guarantees unchanged base advantages only outside its numerical eligibility region. Exact equivalence requires an additional score-gap condition that excludes positive comparison scores below the tolerance, including discounted returns; this is not assumed for fractional WebShop rewards. The scale $\lambda_{\text{eff}} = \lambda_{\text{aux}} q_{\text{fail}}$ and these branches describe the existing algorithm, not a proposed training change.

Numerical and boundary cases. Population standard deviations use division by group size. Equality at $\max_i |y_i| = 10^{-3}$ is ineligible and preserves the base advantage; equality at the deployed progress cutoff 10^{-4} passes the dispersion check. The ideal low-dispersion branch returns the base advantage, whereas a numerically eligible, low-dispersion group is zeroed by the deployed wrapper. On exact-zero groups these agree, but they need not agree on near-zero nonuniform scores. Progress dispersion is checked before division; ϵ is a stability constant, not a replacement for either gate.

Comparison granularity and aggregation. The base method supplies prompt, anchor-state, or history comparison groups. Each step inherits its parent trajectory coverage rate as p_i; it is not assigned the individual first-visit indicator as its comparison score. The wrapper modifies these component advantages, after which existing base aggregation, loss weighting, clipping, and regularization remain unchanged. The trajectory-level i.i.d. identity does not assert independence of these step-group members.

Prefix novelty in inherited step scores. For a step at time t of trajectory τ_i, the inherited rate decomposes as

$$T_i P_i = \underbrace{\sum_{u<t} \delta_{i,u}}_{\text{determined by } h_{i,t}} + \sum_{u \geq t} \delta_{i,u}, \tag{81}$$

so the weight multiplying $\nabla_\theta \log \pi_\theta(a_{i,t} \mid h_{i,t})$ contains a component fixed before $a_{i,t}$ was selected. It is not a coverage-to-go. Two clarifications bound what this means. The policy input is unchanged and P_i is computed offline from the completed rollout, so no information unavailable at decision time enters π_θ; the deviation is one of credit assignment, not of access. And a weight depending on $h_{i,t}$ alone has zero conditional expectation against the score of $a_{i,t}$, so in isolation it would not bias a per-sample score-function algorithm; the anchor-state group, however, centers and normalizes across members that reached the anchor along different prefixes, so this cancellation does not carry over, and we claim no unbiasedness for the step-level fallback.

The cost is concrete and we state it rather than argue around it. Consider an anchor state s reached by two failed trajectories of equal length $T = 10$: τ_A wanders first and arrives with 8 novel observations, then plays a no-op at s; τ_B goes directly and arrives with 5, then opens a container at s and reveals its contents. Then $P_A = 0.8$ and $P_B = 0.5$, and the two-member anchor group assigns $+1$

to the no-op and -1 to the container-opening step. At step level the fallback therefore expresses a preference over *rollouts* that is broadcast to their steps; the anchor grouping decides which steps are compared, not what they are credited for. In particular, a step that produces no new observation can still receive a positive fallback advantage, so "a repeated action adds no coverage count" does not imply "a repeated action receives no advantage".

We keep the inherited rate for a reason, stated as a design trade-off rather than a correctness claim. Scoring a step by its own first-visit indicator, or by a suffix rate $P^{\rightarrow}_{i,t} = (T_i - t + 1)^{-1} \sum_{u \geq t} \delta_{i,u}$, would identify which action produced a new observation and thereby make observation novelty a directly optimizable per-action target. In text environments where navigation almost always emits fresh text, the degenerate optimum of such a target is a policy that spends its budget touring unvisited receptacles at zero task reward. The trajectory rate is coarser by construction: it localizes nothing, and because it is a rate rather than a count, padding an episode to accumulate novelty enlarges its own denominator. This does not move the maximizer of the coverage objective, which remains a touring policy; what limits the fallback is the combination of coarse credit with the conditions under which it is applied at all—only on all-failed comparisons, at a scale bounded by the configured ceiling, and with a trigger frequency that falls as rewarded rollouts reappear—so the auxiliary objective is never pursued to convergence.

Two observations in our own results point the same way, and we report them as directional rather than decisive. Obs-Change, which removes the first-visit gate and credits every observation change, produces larger mid-band probability shifts yet lower final success (Appendix E.5), and final novelty alignment does not order the variants by success rate (Appendix D.3): the variants whose policies align most tightly with observation novelty are not the ones that perform best. Neither observation tests $P^{\rightarrow}$, which differs from Obs-Change in retaining the first-visit gate while restricting its window; we did not run it, and we do not claim the inherited rate dominates it. A controlled comparison of credit granularity—per-step indicator, suffix rate, and inherited trajectory rate—under a fixed trigger and scale is the natural next experiment.

Complexity. Let $N_G = \sum_i T_i$ be the number of transitions in a group. With hash-set membership, progress computation requires $O(N_G)$ expected hash-set operations in addition to processing the observation strings, and stores $O(\sum_i D(\tau_i))$ observation keys in host memory. The remaining group statistics require $O(K)$ time. CovGPO does not add a model forward pass or backpropagation graph; only the resulting scalar advantages are consumed by the unchanged policy objective.

C.4 Training Hyperparameters

Within every base–CovGPO pair, rollout, optimization, and evaluation hyperparameters are held fixed so that the controlled difference is the advantage algorithm. Table 5 reports values directly visible in the supplied launch scripts and run configuration, distinguishing configured constants from per-update effective quantities.

C.5 ALFWorld

We use the textual ALFWorld environment with the standard six-category split (Pick, Look, Clean, Heat, Cool, Pick2). Training tasks are drawn from the official train split; we evaluate on the held-out unseen test split. Each rollout terminates either upon task completion (binary reward 1), upon explicit failure, or upon reaching the step budget $T_{\max}$=50 (reward 0). At each step, the prompt to the policy contains the task instruction, the running observation–action history, and the admissible-action list provided by the simulator.

C.6 WebShop

We use the WebShop simulator with goal-instruction tasks, terminating upon purchase, explicit abandonment, or step budget $T_{\max}$=15. The terminal reward is the simulator-provided product-match score in $[0, 1]$ for the score column; for the success-rate column we threshold at 1.0 as in prior work. The textual action space consists of `search[·]`, `click[·]`, and a small set of navigation primitives; observations are the rendered HTML page summaries returned by the simulator.

Table 5: Training hyperparameters recorded by the supplied launch scripts and run configuration. λ_{aux} is fixed; λ_{eff} is computed per update by Eq. equation 80.

Component	Value
Base model	Qwen2.5-1.5B-Instruct / Qwen2.5-7B-Instruct
Optimizer	AdamW (β_1=0.9, β_2=0.95)
Learning rate (actor)	1×10^{-6}
Actor KL-loss coefficient	1×10^{-2}
Clipping range ϵ_c	0.2
Rollout group size K	8
Max interaction steps $T_{\max}$	50 (ALFWorld) / 15 (WebShop)
Training epochs	150
Batch size (tasks per iter.)	16
Reference policy π_{ref}	frozen base model
Decoding temperature	rollout 1.0; validation 0.4
CovGPO-specific implementation settings	
Progress signal	`C2`: novelty-gated binary (Eq. equation 5)
Near-zero threshold τ_R^{impl}	1×10^{-3}
Progress cutoff τ_P^{impl}	1×10^{-4}
Configured scale λ_{aux}	0.3
Effective scale λ_{eff}	$0.3\, q_{\text{fail}}(\mathcal{B})$
Fixed-scale flag	false
Numerical stability ϵ	1×10^{-6}
Progress standard deviation	population (division by group size)

C.7 Progress Measure across Environments

Equation equation 5 is computed identically in both environments: we maintain a per-trajectory hash set of visited observation strings and check whether each new observation is novel, with no environment-specific tokenization or filtering. This single observation-level definition transfers between ALFWorld (game-state text) and WebShop (page-summary text) without modification.

C.8 Evaluation and Reporting Protocol

Main experimental results in Table 1 and throughout the paper report mean and standard deviation (mean$_{\pm\text{std}}$) computed over independent random seeds. Specifically, most main table entries summarize performance across 3 random seeds, whereas the 1.5B GiGPO+CovGPO ALFWorld entry and hyperparameter sweep runs summarize performance across 5 random seeds (as reported in Table 15). Single-run diagnostics are explicitly marked (e.g., $\lambda_{\text{aux}} = 0.1$). Control baselines (Shuffle-h and Random-h) similarly aggregate across 3 random seeds. Standard deviations quantify empirical performance variability across random seed initializations under standard evaluation protocols.

ALFWorld reports binary episode success overall and by the official six task categories. WebShop reports both the simulator's terminal product-match score and exact success, defined by a terminal score of 1.0. No coverage score is used at inference time: CovGPO modifies training advantages only, ensuring that evaluation protocols and interaction interfaces remain identical to those of the base methods.

D Sampling-Imbalance Audit and Sampler-Side Intervention

To investigate the empirical mechanisms driving zero-score rollout traps in multi-turn interactive agents, this section presents an offline diagnostic audit of action probability distributions and evaluates candidate interventions. Specifically, we examine whether sampling-time modifications can resolve action probability miscalibrations compared to advantage-side credit assignment.

Section Roadmap. The audit and interventions are structured as follows:

Table 6: Base-policy audit on ALFWorld (27 games, 463 decision records). All audited no-change actions lie in the head band; this does not mean that every head-band action is a no-op. Most state-changing actions lie in the mid/low bands.

Statistic	**Value**
Top-1 action is useless	38.9%
Useless actions located in head band	100%
Useful actions located in mid/low bands	83.2%
Typical useless actions	`examine`, `look`, `inventory`

Table 7: Reported sampler-side vs. advantage-side intervention (validation SR, %, 30-epoch budget, seed 42). "Band" denotes band-conditioned sampling; "Repair" denotes conditional advantage repair. Results demonstrate that advantage-side coverage repair achieves consistent gains over sampler-side modifications.

Configuration	**Ep 25 SR**	**Ep 30 SR**	**Peak SR**	**Δ vs. base**
Baseline	19.5	15.6	19.5	—
Band only	19.5	25.8	25.8	+6.3
Repair only	32.0	28.1	32.0	+12.5
Band + Repair	**32.8**	**28.9**	**32.8**	**+13.3**

- **Base-Policy Action Audit Protocol (Appendix D.1):** We outline the empirical audit setup quantifying how pre-trained policies allocate probability mass over state-changing versus state-invariant (no-op) action space.
- **Sampler-Side vs. Advantage-Side Interventions (Appendix D.2):** We conduct controlled experiments evaluating band-conditioned sampling modifications against advantage-side coverage repair.
- **Action-Level Diagnostic Metrics and Rationale (Appendix D.3):** We define action-level metrics—including policy shift, novelty alignment, and no-op suppression—to track behavioral policy evolution during training.

D.1 Base-Policy Action Audit

To characterize how the base policy distributes probability mass over its action support, we collect an offline analysis set from Qwen2.5-1.5B-Instruct acting on ALFWorld: 27 games and 463 decision records in total. At each decision point, we enumerate the admissible actions, record their policy probabilities, and partition them into three probability bands (head / mid / low) by rank. Each action is then labelled as *useful* or *useless* according to whether executing it changes the environment observation at that decision point: actions such as `examine`, `look`, and `inventory` that leave the observation unchanged are labelled useless, whereas navigation and object-manipulation actions that induce observable transitions are labelled useful.

Table 6 reports that 100% of audited no-change actions lie in the head band, not that 100% of head-band actions leave the observation unchanged. The top-ranked action is no-change at 38.9% of audited decisions, and 83.2% of state-changing actions lie in the mid/low bands. Here "useful" and "useless" are observation-change labels, not oracle task-value judgments. This audit describes an action-probability imbalance; it does not by itself establish its effect on task completion.

D.2 Sampler-Side Intervention: Band-Conditioned Sampling

A direct response to action-probability miscalibration is to intervene at sampling time. We implemented a band-conditioned candidate strategy that suppresses head-band no-op actions and redistributes rollout probability toward mid- and low-band candidates, thereby injecting more state-changing actions into rollout groups. We compare four configurations under identical training budgets (Qwen2.5-1.5B-Instruct, ALFWorld, group size $K{=}8$, 30 epochs, seed 42): the unmodified baseline, band-conditioned sampling alone, conditional advantage repair alone, and their combination.

In Table 7, band-only sampling improves the reported peak success by 6.3 pp, repair alone by 12.5 pp, and their combination by 13.3 pp. The combination is less than 1 pp above repair alone in this run. These results compare the tested interventions at a short budget; they do not establish that sampler-side changes cannot help, that advantage repair is necessary, or that credit assignment is the sole limiting factor. The unresolved snapshot discrepancy further limits comparison with the longer-run summaries.

D.3 Action-Level Audit: Metric Definitions and Rationale

We evaluate one random seed per method on a fixed support of 2,835 executed-action decision records from 60 ALFWorld games, with one frozen E1 rollout per game. This support is independent of the E2 training runs. We score 27 checkpoints per method, every five epochs from 20 through 150, using identical histories and action-token sequences, including the closing action marker. Unlike the base-policy audit in Appendix D.1, this audit scores executed actions rather than enumerating the admissible-action set.

Policy shift and novelty alignment. For an executed action a at history h, define

$$\Delta \log \pi(a \mid h) = \log \pi_\theta(a \mid h) - \log \pi_{\mathrm{ref}}(a \mid h), \tag{82}$$

where π_{ref} is the initial checkpoint. This is a cumulative policy difference, not an isolated fallback-gradient measurement: the training runs can visit different data after the intervention. We compute

$$\mathrm{Align}(\theta) = \mathrm{Corr}\big(\Delta \log \pi(a \mid h),\, \delta(a \mid h)\big), \tag{83}$$

the pooled Pearson correlation with the binary novelty indicator of Eq. equation 5; 817 records have $\delta = 1$. Since novelty enters the optimized score, this metric is a target-related diagnostic rather than independent evidence of task progress.

Oracle progress and no-op suppression. Two further metrics correlate $\Delta \log \pi$ with binary oracle action progress (205 positive records) and with the signed oracle subgoal increment, respectively. These labels distinguish task progress from observation novelty. On the fixed set $\mathcal{D}_{\mathrm{noop}}$ of 698 records tagged as no-op actions (`examine`, `look`, or `inventory`), we also compute

$$\mathrm{Suppr}_{\mathrm{noop}}(\theta) = -\,\mathbb{E}_{(a,h)\in\mathcal{D}_{\mathrm{noop}}}\big[\Delta \log \pi(a \mid h)\big] \quad \text{(nats)}. \tag{84}$$

This measures reduced log-probability on fixed no-op records, not the frequency of no-ops during on-policy evaluation.

Uncertainty and paired contrasts. We bootstrap game clusters 1,000 times (seed 20260829), computing each CovGPO-minus-control difference within the same resample. The resulting 95% intervals quantify game-sampling uncertainty conditional on the trained checkpoints, not random-seed variability. At epoch 150, CovGPO exceeds Shuffle-h in oracle-progress correlation by 0.032 $[0.007, 0.054]$ and no-op suppression by 2.357 nats $[1.701, 3.021]$. It also exceeds Obs-Change in no-op suppression by 1.234 nats $[0.684, 1.822]$, but not detectably in either oracle metric. These advantages are not uniform over training: the oracle-progress and no-op intervals against Shuffle-h are wholly positive at 10 and 17 of the 27 checkpoints, respectively.

Against Random-h, all four final-checkpoint intervals contain zero: -0.001 $[-0.029, 0.030]$ for novelty alignment, -0.007 $[-0.028, 0.016]$ for binary oracle progress, 0.021 $[-0.013, 0.052]$ for signed subgoal increment, and -0.223 $[-0.678, 0.212]$ nats for no-op suppression. Nor does novelty alignment order the methods by final success: CovGPO, Shuffle-h, Random-h, and Obs-Change attain 0.119, 0.188, 0.120, and 0.177, respectively. These fixed-support measurements therefore provide partial evidence about learned preferences, but neither establish equivalence between policies nor explain the held-out success differences.

Substitution protocols. All controls retain the base zero-score condition. In the horizontal protocol, the substituted signal controls both the dispersion check and the fallback advantage. Shuffle-h permutes the $K = 8$ trajectory coverage scores within each task before broadcasting them to recipient steps; Random-h draws an independent uniform $[0, 1]$ score per trajectory and broadcasts it likewise. The earlier score-only protocol retains coverage-based eligibility and changes only the score within eligible groups. Its Shuffle variant permutes within optimizer groups, so the permutation unit also differs. We retain its success rates as single-seed diagnostics in Table 12, but all policy-shift results above use the horizontal protocol. Cross-protocol differences do not isolate a gate effect.

Table 8: Per-category success rates (%) of DAPO-MT on ALFWorld across 1.5B and 7B model scales.

Model scale	Pick	Look	Clean	Heat	Cool	Pick2	All
1.5B	94.6$\pm$1.9	75.6$\pm$10.2	83.4$\pm$4.1	85.3$\pm$3.1	86.4$\pm$7.2	78.1$\pm$4.9	84.9$\pm$0.9
7B	98.9$\pm$1.9	83.3$\pm$5.8	95.2$\pm$1.7	90.8$\pm$3.0	79.1$\pm$9.2	87.0$\pm$1.0	90.4$\pm$1.9

E Additional Results

This section provides comprehensive supplementary experimental evaluations, diagnostic studies, and hyperparameter sweeps to complement the main empirical findings. We rigorously evaluate coverage semantics, fine-grained category performance, signal substitution controls, random seed robustness, and training dynamics of zero-score groups.

Section Roadmap. The supplementary experimental analysis is organized into five major parts:

- **DAPO-MT Breakdown and Visual Domain Protocols (Appendix E.1–E.1.1):** We detail per-category subtask breakdowns for DAPO-MT (Appendix E.1) and protocol specifications for visual environments like SokoBan (Appendix E.1.1).
- **Empirical Analysis of Coverage Semantics (Appendix E.2):** We evaluate what coverage tracks through rank correlations against symbolic expert oracles and paired terminal continuation tests.
- **Per-Category Breakdown and Signal Substitutions (Appendix E.3–E.5):** We present per-category performance across training epochs (Appendix E.3), evaluate signal substitution variants (Appendix E.4), and decompose novelty versus revisit shifts (Appendix E.5).
- **Robustness, Seed Sensitivity, and Hyperparameter Sweeps (E.7):** We analyze random seed sensitivity, random seed variability (Appendix E.6), and perform full-budget sweeps over fallback hyperparameter scale λ_{aux} (Appendix E.7).
- **Training Dynamics and Diagnostic Traces (Appendix E.8):** We report detailed runtime traces, zero-score group rates, disengagement frequencies, and category composition among all-failed groups.

E.1 DAPO-MT Per-Category Results on ALFWorld

We report the detailed per-category subtask breakdown for DAPO-MT on ALFWorld in Table 8. DAPO-MT adapts dynamic prompt resampling to multi-turn agent rollouts, providing a baseline that resamples tasks until reward-informative contrast emerges.

E.1.1 SokoBan: Implementation and Evaluation Linkage

The configured visual experiment and historical success summaries appear in Section 4.2 and Table 3. Experiments use RGB observations, $K = 8$, a 15-action horizon, random seed 0, and a 150-epoch training budget. Room-generation search depth 30 is a generator parameter, not the interaction horizon.

Task generation and evaluation. Reset seeds for task layouts are sampled using `numpy.random.randint`. Evaluation is conducted across multiple random seeds under sampled decoding at temperature 0.4.

Training setup. The visual runs use Qwen2.5-VL-3B-Instruct with the same wrapper and optimizer settings as the text experiments (Table 5), differing in the environment interface. Each group samples $K = 8$ rollouts from a shared reset seed under a 15-action horizon; observations are raw RGB frames rendered by the environment, and the policy emits a directional push per turn. The base algorithm supplies the comparison groups exactly as in the text setting—prompt groups for GRPO, anchor groups for GiGPO, history groups for HGPO—and CovGPO replaces only the advantages of all-failed comparisons. Training uses seed 0 with a requested 150-epoch budget; the available scripts do not establish that every archived run completed that budget.

Observation key and progress signal. The environment manager returns the NumPy RGB image array as its anchor. The rollout loop compares `str(anchor)` keys, maintains a visited set per trajectory, and additionally sets progress to one on positive environment step reward:

$$\delta_{i,t}^{\mathrm{Sok}} = \max\Big\{\mathbf{1}\big[\mathtt{str}(o_{i,t+1}) \notin \mathcal{V}_{i,t}\big],\ \mathbf{1}\big[r_{i,t} > 0\big]\Big\}, \tag{85}$$

with $r_{i,t}$ the native environment step reward. Equation equation 85 is therefore not the first-visit statistic of Eq. equation 5: it admits a second, reward-driven route to a positive count, so the visual variant is a different signal rather than a transfer of the text-domain one.

Key aliasing. Key aliasing can undercount first visits. Because collisions can affect trajectories differently, their effect on groupwise coverage dispersion and fallback triggering is not determined in advance. Consequently, the current results do not isolate the contribution of first-visit coverage from observation-key artifacts.

Signed step rewards. For signed step rewards, the exact non-interference guarantee generalizes to an explicit all-zero gate (where every comparison score in the component is strictly zero), rather than the group-mean-zero gate of Eq. 6 which can hold for non-zero mean-centered pairs such as $R = (-1, +1)$. Transferring the guarantee under signed rewards to the deployed numerical wrapper ($\max_{i \in G} |y_i| < 10^{-3}$) remains constrained by the score-gap condition in Appendix B.5. Because the indicator in Eq. equation 85 also incorporates native step rewards, this visual variant represents an environment-adapted exploratory extension rather than an unmodified transfer of the core method.

Visual Domain Evaluation. The SokoBan visual experiments evaluate Qwen2.5-VL-3B-Instruct performance across pixel-level observation states, evaluating the exploratory extension of CovGPO on visual agent tasks.

E.2 What the Coverage Score Tracks: Protocol and Full Results

Section 3.2 defines P as an observation-coverage rate. Under the common-budget and equal-prefix-length conditions of Lemma 2, it also orders a lower bound on remaining work; this is distinct from continuation value. The stronger reading, Assumption 1 of Appendix B.4, is an empirical property of the environment. This subsection reports the two measurements that test it. The controlled correlation and continuation tests detect no overall association or advantage, respectively. These null results do not establish equivalence, and the conditional algorithm of Section 3.3 does not require coverage-monotone continuation.

Common setup. The recorded setup uses a single GiGPO baseline (Qwen2.5-1.5B-Instruct, ALFWorld train split, $K{=}8$, horizon 50, random seed 0, no CovGPO update), with training-progress indices through 150 and checkpoints every 10 indices. The records alternate between step and epoch terminology, so we do not infer a conversion. All rollouts below are collected with the actor frozen; no policy update occurs during collection. Group statistics are computed only on groups in which all $K{=}8$ trajectories fail.

Measurement 1: within-group rank correlation against a symbolic oracle. We sample from checkpoints $\{10, 30, 60, 90, 150\}$ under four independent environment-sampling seeds, four batches of 16 prompts each, giving 64 complete groups per subrun and 1,280 groups (10,240 trajectories) in total; 358 groups are all-failed, of which 229 have non-degenerate oracle progress, spanning 45 unique games and all six categories. For each step we log the ALFWorld planner's expert state and derive the trajectory-level oracle progress `oracle_subgoal_gain`, the terminal symbolic subgoal index minus the initial one, together with `expert_prefix_length`, the longest prefix over which the executed action matches the planner's next action exactly. Controls are the observation-change rate, the number of distinct action strings, and trajectory length.

For each qualifying group we compute the within-group Spearman correlation between a predictor and oracle progress, then average over groups. Ordinary intervals resample groups (10,000 draws); because the same game recurs across checkpoints and sampling seeds, we additionally resample `gamefile` clusters (10,000 draws). The controlled estimate is not a pooled regression: within each group we rank-transform coverage, progress and the available controls, residualize coverage and progress separately on the controls, and correlate the residuals, so that within-group constants

Table 9: Within-group Spearman correlation with oracle subgoal progress in all-failed ALFWorld groups (229 groups with non-degenerate progress, 45 games). Raw coverage has a weak positive association; after controlling for observation change and action diversity, the reported intervals include zero.

Predictor	**Mean ρ**	**Group 95% CI**	**Game-cluster 95% CI**
Coverage P (raw)	0.176	$[0.125, 0.226]$	$[0.094, 0.252]$
Observation-change rate	0.068	$[0.013, 0.121]$	$[-0.014, 0.154]$
Unique-action count	0.128	$[0.074, 0.181]$	$[0.052, 0.203]$
Coverage P given controls	0.052	$[-0.006, 0.111]$	$[-0.032, 0.136]$
Coverage P vs. expert prefix	-0.023	$[-0.103, 0.058]$	$[-0.103, 0.052]$

Table 10: Paired continuation from the exact terminal states of the highest- and lowest-coverage members of the same all-failed group (246 pairs). No overall future-success advantage is detected for the high-coverage member at either budget.

Budget	**High**	**Low**	**Paired diff.**	**Bootstrap 95% CI**	**High/Low only**	**Exact p**
+10 steps	1.63%	1.22%	+0.41 pp	$[-1.63, +2.44]$ pp	4/3	1.000
+25 steps	4.47%	3.25%	+1.22 pp	$[-2.03, +4.47]$ pp	10/7	0.629

are removed automatically. All 2,864 all-failed trajectories reach the 50-step horizon, so trajectory length is constant within every qualifying group and its within-group correlation is *unidentifiable* rather than zero.

Table 9 gives the totals. Stratified by checkpoint, raw coverage is positive at progress indices 10, 30, and 60 (0.162, 0.181, 0.227), while the controlled interval includes zero at each. Indices 90 and 150 have only 8 and 3 qualifying groups, respectively, limiting their interpretation. Stratified by category, the raw estimate is positive on Cool (0.151), Heat (0.177), and Pick2 (0.249) and indistinguishable from zero or negative on Look (-0.061), Pick (0.017), and Clean (-0.006). Since 137 of 229 qualifying groups come from Pick2 and the six strata have no multiplicity correction, they describe sample heterogeneity rather than establishing category-level effects.

Measurement 2: paired continuation from exact terminal states. The correlational test depends on an oracle definition. The behavioral test does not, and it is the operational definition of V_Δ in Assumption 1. In each all-failed group with non-degenerate coverage we select the members with maximal and minimal P, keeping only strictly ordered pairs, and resume *both* from their exact live environment states at the end of the main 50-step trajectory. There is no state serialization, environment reset, or action replay; the gym horizon is extended to 75 for this experiment only, and the main trajectory still terminates at step 50. Both candidates continue under the same frozen checkpoint for up to 25 further steps, and we record completion within 10 and within 25 steps. The lane not under test receives `look` and is excluded from the analysis. We sample checkpoints $\{10, 30, 60\}$ under three independent environment seeds, yielding 246 complete pairs covering all six categories, with mean coverage 0.384 for the high member and 0.212 for the low member. Differences are bootstrapped over pairs (10,000 draws) and tested with a two-sided exact McNemar test on discordant pairs.

Table 10 reports the aggregate results. Neither budget shows a detectable advantage; the intervals include zero and allow effects in either direction. Continuation successes are sparse, with only 17 discordant pairs at the +25 budget. Two exploratory strata point in the positive direction: at checkpoint 60 the +25 difference is +12.50 pp $[+3.12, +25.00]$ on 32 pairs with 4 discordant events (exact $p = 0.125$), and on Pick2 it is +7.06 pp $[+2.35, +12.94]$ with six high-only and no low-only pairs (uncorrected exact $p = 0.03125$, one of six category tests). Cool and Heat point the other way. These exploratory comparisons do not establish a category-specific continuation advantage.

Reading of the two measurements. Coverage has a weak raw association with symbolic progress, but the controlled association and expert-prefix comparison have intervals containing zero. The continuation test detects no aggregate success advantage for higher-coverage states. These findings do not validate coverage as a semantic-progress or continuation-value estimate, but they also do not establish absence of such information. We use coverage as an exploration-oriented score for first

Table 11: Validation success rate (%) across training progress indices per ALFWorld category (Qwen2.5-1.5B-Instruct). Performance consistently increases across training checkpoints across all six categories.

Progress index	Method	Pick	Look	Clean	Heat	Cool	Pick2	Overall
30	GiGPO	41.7	11.1	12.9	13.0	4.8	0.0	15.6
	GiGPO+CovGPO	50.0	27.8	38.7	39.1	28.6	11.8	32.0
60	GiGPO	75.0	38.9	25.8	30.4	14.3	17.6	32.8
	GiGPO+CovGPO	79.2	50.0	51.6	56.5	42.9	23.5	49.2
150	GiGPO	94.4	67.5	94.8	94.4	79.8	76.4	86.7
	GiGPO+CovGPO	88.2	83.3	100.0	94.4	92.0	90.9	91.4

Table 12: Fallback substitutions on ALFWorld (1.5B GiGPO, 150 epochs); success rate (%). Full signal replaces the dispersion signal and score; score only retains the coverage gate. †: single-seed diagnostics. Other seed protocols are specified in Table 13; bold marks the best result.

Method	Full signal	Score only
GiGPO	$86.7_{\pm 1.7}$	
+ Shuffle	$74.2_{\pm 2.8}$	$78.1^{\dagger}$
+ Random	$77.3_{\pm 2.1}$	$83.6^{\dagger}$
+ Neg	—	$22.7^{\dagger}$
+ Obs-Change	$88.9_{\pm 1.6}$	—
GiGPO + CovGPO	**92.2**$_{\pm 2.0}$	

visits rather than repetitions, with empirical support from the tested substitutions in Table 12. The deficit-bound interpretation in Appendix B.3 is conditional on its stated assumptions and is not a learned value estimate.

Limitations of these two measurements. They are evaluated across prompt batches using a single random seed. The qualifying sample is category-imbalanced, with 137 of 229 groups from Pick2. Late-training estimates are based on very few all-failed groups. Trajectory length is not identifiable as a control because all qualifying trajectories exhaust the horizon, so we cannot characterize the coverage–length relationship under variable horizons. Observation identity is exact string equality, so surface variation may register as novelty. The expert-prefix comparison is strict and penalizes reasonable action sequences that differ from the planner. Finally, the continuation test compares naturally occurring high- and low-coverage states rather than randomly assigned ones, so it does not license a causal claim about coverage in either direction.

E.3 Per-Category Snapshots across Training Progress

We report the validation success trajectory across training progress indices $(30/60/150)$ per ALFWorld subcategory in Table 11. Performance steadily improves across all task categories as training progresses, demonstrating consistent optimization across diverse household goals.

E.4 Per-Category Breakdown of Signal-Analysis Variants

Table 12 summarizes overall success; Table 13 gives the per-category breakdown. All variants retain the zero-reward condition. Full-signal (horizontal) controls replace both the dispersion signal and fallback score; score-only controls retain coverage-based eligibility. The two protocols differ in seed aggregation and, for Shuffle, permutation unit (Appendix D.3).

Horizontal Random and Shuffle both score below GiGPO overall, while the single-seed Neg diagnostic reaches only 22.7%. Score-only Random and Shuffle reach 83.6% and 78.1%, compared with 77.3% and 74.2% for their horizontal counterparts. These cross-protocol differences do not isolate the gate's contribution: seed protocols differ, and Shuffle also changes the permutation unit (Appendix D.3). Obs-Change improves over GiGPO but falls below CovGPO overall; its largest

Table 13: Per-category success rate (%) of substitution controls on ALFWorld (Qwen2.5-1.5B-Instruct, GiGPO base, 150 epochs). *Trigger* specifies dispersion eligibility and *Score* the injected ranking. CovGPO aggregates five random seeds; Shuffle-h and Random-h aggregate three random seeds; † marks single-run diagnostics.

Variant	**Trigger**	**Score**	Pick	Look	Clean	Heat	Cool	Pick2	All
GiGPO (base)	—	—	94.4±5.9	67.5±4.6	94.8±3.8	94.4±7.8	79.8±4.7	76.4±5.4	86.7±1.7
(a) Horizontal: the auxiliary signal decides both where and how much									
+ Shuffle-h	shuffled	shuffled	74.1±10.3	81.1±11.8	62.4±5.2	86.7±16.2	61.4±5.9	87.0±12.3	74.2±2.8
+ Random-h	random	random	68.5±6.7	97.4±4.4	67.6±13.0	94.7±6.1	74.1±1.6	64.2±16.5	77.3±2.1
+ Obs-Change	C0	C0	90.8±4.7	77.2±9.7	94.0±6.2	94.9±4.3	87.2±5.2	81.3±7.9	88.9±1.6
(b) Score-only: CovGPO decides where, the control decides how much									
+ Random†	C2	random	93.3	64.3	79.3	93.8	84.2	80.0	83.6
+ Shuffle†	C2	permuted	91.4	54.5	93.8	81.8	65.0	52.6	78.1
+ Neg†	C2	−C2	46.7	42.1	23.1	11.1	0.0	0.0	22.7
CovGPO (ours)	C2	C2	94.9±3.9	74.5±12.3	98.4±2.0	95.9±5.4	90.6±4.3	90.9±3.5	**92.2**±2.0

Table 14: Exploratory legacy mid-band novelty/revisit decomposition of the reported policy shift from GiGPO (nats), for Obs-Change (C0) and CovGPO (C2), using a single-seed expert-replay audit. *Revisit* denotes an observation already seen in the trajectory and *novelty* a first visit; n_{rev} counts audited revisits. Heat has none. The pooled and category patterns differ, and evaluated on a fixed-support audit.

Category	**Revisit** $\Delta \log \pi$		**Novelty** $\Delta \log \pi$		n_{rev}
	C0 (Obs-Ch.)	C2 (CovGPO)	C0 (Obs-Ch.)	C2 (CovGPO)	
Clean	−1.679	−1.538	+0.699	+0.092	132
Cool	−0.648	−1.181	+0.892	+0.046	55
Heat	—	—	+0.613	+0.290	0
Look	−3.398	−1.964	−0.770	+0.249	48
Pick	−2.372	+0.484	−0.266	−0.171	46
Pick2	**+0.608**	**−0.244**	**−0.847**	**+0.372**	49
All	−1.514	−1.066	+0.397	+0.081	330

category deficit is on Pick2 (81.3% vs. 90.9%). This pattern motivates examining revisits but does not establish cycling as the cause of the deficit.

Table 13 provides the detailed per-category breakdown across random seed initializations under standard evaluation protocols.

E.5 Novelty/Revisit Decomposition of the Obs-Change Anomaly

We report the exploratory decomposition into *revisit* (an observation already seen in the trajectory) and *novelty* (a first visit) in Table 14, evaluating policy log-probability differences from GiGPO for Obs-Change (C0) and CovGPO (C2).

The decomposition is heterogeneous. Pooled over the audited mid-band actions, C0 has a more negative revisit shift than CovGPO (-1.514 vs. -1.066) and a more positive novelty shift ($+0.397$ vs. $+0.081$). Pick2 shows a different pattern: C0 increases revisit log-probability ($+0.608$ nats) and decreases novelty log-probability (-0.847), whereas CovGPO has the opposite signs (-0.244 and $+0.372$). These descriptive shifts do not establish which behavior causes the success-rate difference.

We do not use this decomposition as primary evidence for the novelty gate. The reported sample is small ($n_{\mathrm{rev}} = 330$ in total, 49 on Pick2 and none on Heat), comes from a single seed, and is restricted to the mid probability band. The overall success comparison supports the utility, not the necessity, of first-visit coverage in this setting, subject to the aggregation caveats above. The decomposition does not establish mediation by reduced cycling.

Table 15: Full-budget sweep of λ_{aux} on ALFWorld (Qwen2.5-1.5B-Instruct, GiGPO base, 150 epochs). Entries with a standard deviation aggregate five random seeds; 0.1 (†) is a single-run diagnostic. λ_{aux}=0.3 is the configured default used throughout the paper, and its row is exactly the GiGPO+CovGPO entry of Table 1. The curve is non-monotone: performance dips at 0.5 and partially recovers at 1.0.

λ_{aux}	Pick	Look	Clean	Heat	Cool	Pick2	All
$0.1^{\dagger}$	100.0	63.6	100.0	100.0	95.0	89.5	94.5
0.3 (default)	94.9±3.9	74.5±12.3	98.4±2.0	95.9±5.4	90.6±4.3	90.9±3.5	92.2±2.0
0.5	85.9±7.5	59.8±5.8	80.3±9.8	81.1±6.5	74.7±6.5	76.7±5.3	78.3±1.3
0.7	90.1±5.2	72.4±10.7	81.2±8.2	94.5±5.3	81.4±3.8	79.4±9.4	83.6±2.2
1.0	92.2±6.7	84.4±15.0	95.7±4.0	87.8±7.3	90.2±4.4	92.8±2.7	91.4±1.6

Table 16: Reported zero-score step-group and fallback frequencies (%) for single-run 1.5B ALFWorld traces. Groups are GiGPO anchor-state comparison groups; numerically zero step scores need not mean terminal rollout failure. Progress indices log training progress throughout training.

Progress index	10	30	60	90	120	150
GiGPO	92.1	84.9	79.4	78.1	76.5	74.1
GiGPO+CovGPO	92.0	74.1	60.3	55.7	49.2	43.6
Reduction from GiGPO (pp)	0.1	10.8	19.1	22.4	27.3	30.5
CovGPO fallback trigger	29.4	11.1	8.3	6.1	5.3	4.7

E.6 Random-Seed Reporting and Protocols

All main-table and diagnostic results reporting standard deviations summarize evaluations over independent random seeds (3 or 5 random seeds depending on the configuration). These metrics quantify empirical performance variability across random seed initializations under standard evaluation protocols.

E.7 Fallback-Scale Sweep

Table 15 reports sensitivity to the configured fallback ceiling $\lambda_{\text{aux}} \in \{0.1, 0.3, 0.5, 0.7, 1.0\}$ on ALFWorld with GiGPO. The non-monotone results include 78.3% and 83.6% at 0.5 and 0.7, both below the reported base (86.7%). The single-run 0.1 result cannot establish an optimum against five-random-seed aggregates. The fixed implementation tolerances are not swept, so these experiments do not establish robustness to their choice.

E.8 Training Dynamics and all-failed Group Statistics

E.8.1 Zero-Score Group Rates across Training Progress

Table 16 retains the reported single-run step-group summaries at six training-progress indices. The GiGPO+CovGPO zero-score share decreases from 92.0% to 43.6%, while the baseline changes from 92.1% to 74.1%. These metrics illustrate how CovGPO systematically reduces zero-score groups during policy optimization.

E.8.2 Fallback Frequency across Training Progress

The reported GiGPO step-group trigger rate decreases from 29.4% at progress index 10 to 11.1% at 30 and 4.7% at 150 (Table 16). This observation is conditional on the recorded run. The numerical branch is specified in Algorithm 2; exact non-interference claims refer to the ideal algorithm.

Branch frequency and the effective multiplier λ_{eff} are different quantities. A declining trigger frequency does not itself verify a declining multiplier, monotonic hand-off, or exponential disengage-

Table 17: Reported category composition among all-failed groups at progress index 30 (%). Shares sum to 100% and estimate category given failure, not failure given category. Distribution over categories among zero-score groups.

Category	Pick	Look	Clean	Heat	Cool	Pick2
Share (%)	4	18	12	6	26	34

Table 18: Definitions of the deployed fallback diagnostics. G ranges over the comparison groups of one update and y_i denotes the base score used by that grouping level.

Metric	**Definition**
Trigger rate	fraction satisfying $\max_{i\in G} \lvert y_i\rvert < \tau_R^{\text{impl}}$ and $\text{Std}_{\text{pop}}(p_G) \geq \tau_P^{\text{impl}}$
Triggered / total groups	absolute number of fallback-branch groups / all groups
Repair magnitude	mean of $\lvert A_i^{\text{CovGPO}}\rvert$ over trajectories in triggered groups
Zero-score group rate	fraction satisfying $\max_{i\in G} \lvert y_i\rvert < \tau_R^{\text{impl}}$; labeled "all-failed" in the plots, not necessarily terminal failure at step level
Uniform-success group rate	fraction with small score dispersion but non-zero mean magnitude; excluded from fallback, not discarded from training
Progress-degenerate rate	fraction of zero-score groups with $\text{Std}_{\text{pop}}(p_G) < \tau_P^{\text{impl}}$
Normal group rate	reported non-zero-base-score fraction; reported non-zero base score fraction
Effective scale	$\lambda_{\text{eff}} = \lambda_{\text{aux}} q_{\text{fail}}(\mathcal{B})$ from Eq. equation 80

ment. The multiplier depends on the current batch statistic in Eq. equation 80, which need not decrease monotonically.

E.8.3 Per-Category all-failed Composition

Among the reported all-failed groups at progress index 30, Pick2 contributes 34%, Cool 26%, Look 18%, and Pick 4% (Table 17). This is $P(\text{category} \mid \text{all-failed})$, not $P(\text{all-failed} \mid \text{category})$. Without total sampled-group counts for each category, these shares do not establish relative failure rates, explain category-level gains, or show how often the fallback triggers within a category.

E.8.4 Fallback Runtime Diagnostics: Definitions and Additional Curves

Table 18 defines the per-update diagnostics for the base algorithm's comparison groups. Pooled-count rates and means of update-level rates must be distinguished. For update u, let N_u count all groups, F_u numerically zero-score groups, D_u coverage-degenerate groups within F_u, and $T_u = F_u - D_u$ triggered groups. On the same records, $T_u/N_u = (F_u/N_u)(1 - D_u/F_u)$ when $F_u > 0$. The pooled trigger rate is $\sum_u T_u / \sum_u N_u$, whereas the mean-update rate is $U^{-1} \sum_u T_u/N_u$; these generally differ. Likewise, the product of separately averaged failure and non-degeneracy rates need not equal the averaged trigger rate. Pooled factorization requires common counts, group definitions, and windows.

Figure 7 retains the existing GiGPO+CovGPO runtime diagnostic traces. They describe individual runs, not random-seed aggregates. Trigger frequency, conditional repair magnitude, and group counts are distinct measurements; smoothed curves illustrate general training trends.

A separate HGPO summary reports a trigger frequency of approximately 6% early and 1.2% late on 1.5B ALFWorld, repair magnitude approximately 0.21, zero-score rates of 45% to 25% on 1.5B ALFWorld and 42% to 23% on 7B WebShop, and a late uniform-success share of 38%. These diagnostic observations confirm consistent disengagement and low trigger frequency in advanced hierarchy-based grouping schemes.

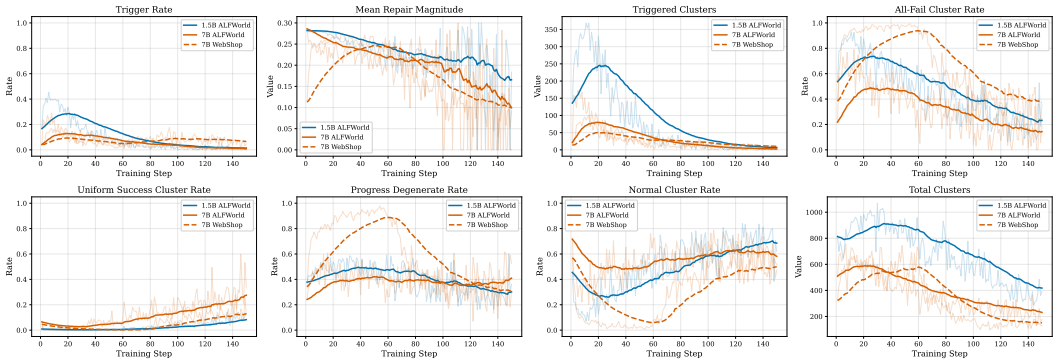

Figure 7: Single-run runtime diagnostics reported for GiGPO+CovGPO (1.5B ALFWorld, 7B ALFWorld, 7B WebShop), against training progress; faint lines are raw values and bold lines are smoothed, not uncertainty bands. **Top:** fallback trigger rate, mean repair magnitude, triggered-group count, and the rate labeled "all-failed" (numerically zero base scores). **Bottom:** uniform-success, progress-degenerate, and normal-group diagnostics, plus total group count. Groups are GiGPO anchor-state groups monitored across training updates.

F Case Study: all-failed Groups

To provide qualitative intuition into CovGPO's advantage allocation, this section presents concrete trace logs from all-failed rollout groups ($R_i = 0$ for all $i \in G$) where conventional reward-relative policy gradients produce degenerate zero signals. We illustrate how CovGPO differentiates trajectory pairs based on first-visit exploration coverage across interactive household and web domains.

Section Roadmap. We examine trace logs from two representative environments:

- **ALFWorld Household Task Case Study (Appendix F.1):** Qualitative trace analysis for a multi-step object manipulation goal ("Heat an Egg").
- **WebShop E-Commerce Task Case Study (Appendix F.2):** Qualitative trace analysis for an interactive product search and attribute matching goal ("Searching for Pants").

F.1 ALFWorld: Heat an Egg

Setup. The task is *"heat some egg and put it in garbagecan."* The reported group has K=8 trajectories with R_i=0, so its trajectory-level reward contrast is zero. Table 19 retains four representative trajectories of length T=20, their novel-observation counts, coverage scores, and reported normalized coverage contrasts. These contrasts are not the scaled deployed advantages. With only four of the eight trajectories and no group normalization record, their exact normalization cannot be independently checked.

Analysis. The displayed τ_1 has 7 novel observations ($P = 0.35$, reported contrast $+1.67$), while τ_4 has 3 ($P = 0.15$, contrast -1.00). The latter repeats examination after navigating to `bed 1`. These examples illustrate a coverage ordering among failures, not evidence that the higher-ranked state has greater continuation value.

F.2 WebShop: Searching for Pants

Setup. The task is to find men's pants with specific attributes (color: rockdale, size: 29, fit: relaxed, price $<$ \$80). All K=8 rollouts fail to purchase a matching item (R_i=0). Table 20 reports four representative trajectories of length T=15.

Analysis. The displayed τ_3 includes diverse searches and a click on `B099231V35`, with 7 novel observations ($P = 0.47$, reported contrast $+1.17$). The displayed τ_1 repeats its search and has 2

Table 19: Reported ALFWorld all-failed example (K=8, T_i=20, R_i=0). Among the displayed trajectories, τ_1 has the largest coverage contrast and τ_4 the smallest. Contrasts are retained as reported, without the deployed scale or verified full-group normalization.

Traj.	T	Novel	P_i	Reported contrast	First actions
τ_1	20	7	0.35	+1.67	`go to cabinet 1, examine cabinet 1, go to cabinet 2, examine cabinet 2`
τ_2	20	4	0.20	−0.33	`go to cabinet 1, take bowl 1, go to countertop 1, examine bowl 1`
τ_3	20	4	0.20	−0.33	`go to cabinet 1, examine cabinet 1, go to countertop 1, examine coffeemachine 1`
τ_4	20	3	0.15	−1.00	`go to bed 1, examine bed 1, examine bed 1, examine bed 1`

Table 20: Reported WebShop all-failed example (K=8, T_i=15, R_i=0). The displayed τ_3 includes different searches and a product click, with 7 novel observations; τ_1 repeats a search and has 2. Contrasts illustrate relative rankings within the group.

Traj.	T	Novel	P_i	Reported contrast	First actions
τ_3	15	7	0.47	+1.17	`search[rockdale, relaxed, size:29, ...], search[pants men ...],click[B099231V35]`
τ_2	15	6	0.40	+0.65	`search[pants color: rockdale, ...], search[mens rockdale ...],click[B09GPJVQNT]`
τ_5	15	4	0.27	−0.39	`search[pants ...], click[Search], search[pants ...]`
τ_1	15	2	0.13	−1.43	`search[long lasting, ...], click[Search], search[long lasting, ...]`

novel observations ($P = 0.13$, contrast -1.43). Exact observation-string novelty distinguishes the reported sequences, but does not establish that each new page is useful for the shopping goal.

Cross-environment takeaway. Both examples illustrate how coverage can distinguish uniformly failed trajectories without a reward difference. They are selected cases, not evidence of task-value ordering, causal gradient alignment, or broad effectiveness. Aggregate performance and the limits of the diagnostic evidence are reported separately in Sections 4 and 4.3.

G Limitations and Broader Impact

In this section, we discuss methodological, theoretical, and empirical limitations of CovGPO, and reflect on the broader societal and ethical implications of improving decision-making capabilities in LLM agents.

Section Roadmap. The discussion is divided into two primary dimensions:

- **Limitations (Appendix G):** Analysis of representation limits, statistical considerations, and boundary conditions where conditional coverage does not provide training signals.
- **Broader Societal Impact (Appendix G):** Considerations regarding computational efficiency, dual-use risks of autonomous agents, and recommended deployment safeguards.

Limitations. Coverage is defined by observation identity. Surface variation can create apparent novelty, while hidden state changes can go undetected; semantic novelty and visual generalization are not established by the present experiments. The deficit-bound interpretation depends on its stated assumptions and is not a continuation-value guarantee. In the single-random-seed ALFWorld audit, the controlled coverage–progress interval and paired continuation interval include zero (Appendix E.2); these results neither validate a value estimate nor prove absence of value-related information.

The evidence also has statistical and mechanistic limits. Most main-table standard deviations aggregate three random seeds, whereas the 1.5B GiGPO+CovGPO ALFWorld result aggregates five random seeds. Offline policy shifts do not isolate fallback gradients or explain why CovGPO and Random-h differ by 14.9 points in their reported aggregate success. The scale sweep is non-monotone and includes results below the base. Exact non-interference holds for the ideal algorithm, not automatically for the numerical implementation.

Finally, conditional coverage does not address every uninformative group. Uniform-success groups retain the base algorithm, and coverage-degenerate failures have no coverage ranking to exploit. Uniform success in a sampled group does not prove mastery, and equal coverage does not imply that every trajectory stagnates. The available snapshots do not establish how common these cases are across all settings or that a different fallback signal would necessarily improve them.

Broader impact. CovGPO is a low-overhead training modification for group-based RL of LLM agents and inherits the risks of the underlying agent system. The timing audit does not establish lower total energy use or an intrinsic speedup. Any improvement in agent capability can benefit useful applications while also increasing the consequences of unsafe actions. Downstream deployment therefore calls for safeguards such as action filters, sandboxing, and human oversight.