LIVE-BROWSER-AGENT: REAL-WORLD CENTRIC FOUNDATION LONG-HORIZON BROWSER AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Training a browser agent for deployment requires assigning credit from a sparse terminal outcome to the few critical steps that determined it. Existing pipelines supervise on successful demonstrations and optimize online with group-relative objectives that assign a uniform trajectory-level advantage to every step. Methods resolving credit below the trajectory level either intervene on the environment or match steps via recurring observations, neither of which is reliable in live browsing sessions. We propose a two-stage training framework to address this. First, Recovery- and UI-specialized Curriculum Supervised Fine-Tuning (RUIC-SFT) treats recovery data as state-conditioned supervision, introducing it only after the conditioning state becomes separable; this significantly improves erroneous-step recovery while reducing redundant actions per task. Second, Divergence-Attributed Online Group Relative Policy Optimization (DAO-GRPO) derives unbiased step-level advantages from prefix-conditioned leave-one-out baselines within each rollout group, requiring no environment resets or per-step model calls. Building on these stages, we develop the Live-Browser-Agent family (4B, 9B, and 27B). Our 27B model attains the highest average Pass@1 among open-weight models on three online browser benchmarks (82.3%), on par with Kimi-k3 and Qwen3.8-Max despite using roughly 90× fewer parameters, and improves its Qwen3.5-27B initialization on three general agentic benchmarks, indicating that browser-centric training does not erode general tool use.

1 INTRODUCTION

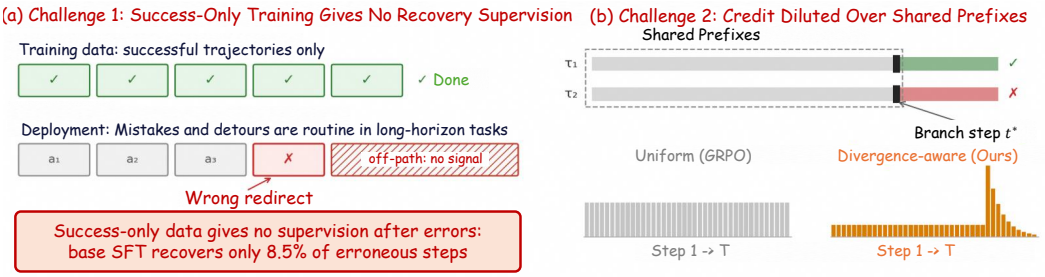
Deployed browser agents must make dozens of decisions on live pages, yet training data and benchmarks remain concentrated on short, successful, English-centric interactions. Our bilingual BrowserBench quantifies this gap: across 350 real-web tasks averaging 37.9 human-demonstrated steps, even the strongest proprietary agents fail on a substantial fraction, and strong supervised baselines drop sharply beyond 50 steps.

These failures share a structural cause: web workflows contain few outcome-determining decisions; most steps are page-constrained or locally reversible (Deng et al., 2023), and per-step supervision is unavailable without expensive resets. We instead exploit outcome divergence among sibling rollouts sharing an action prefix—under terminal feedback, early errors surface many steps late, making decisive-step credit assignment the central difficulty of long horizons (Figure 1).

The first challenge is supervision composition. Public browser-agent corpora are dominated by successful trajectories and rarely contain verified recovery behavior. Recovery supervision is state-conditioned: without an execution prior from D_g, the policy tends to fit the marginal and emit corrective actions regardless of whether an error occurred, leading to *over-hedging*. The second challenge is online optimization. Rollouts often share long action prefixes and diverge at only a few decisive steps, yet neither environment intervention nor observation matching is reliable on live pages. The third challenge is evaluation. Existing benchmarks are mostly short-horizon and English-centric, so many errors surface immediately instead of propagating over long executions.

We address these challenges with a dataset, two training stages, and a benchmark. Live-Browser-Set (Section 4) provides step-annotated trajectories with verified corrections. RUIC-SFT (Section 5) introduces recovery supervision only after its conditioning state becomes separable. DAO-GRPO (Section 6) estimates step-level advantages via prefix-conditioned leave-one-out baselines under

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.



(c) Challenge 3: Bilingual Long-Horizon Real-Web Tasks are Underrepresented

Benchmarks	Online	Tasks	Websites	Language	Benchmark Evaluation Method	Long-Horizon Evidence
WebArena	✗	812	5	English	Execution-based programmatic verification	Moderate Horizon (Avg. Step = 10)
WebVoyager	√	643	15	English	LLM-as-a-Judge (GPT-4V)	Moderate Horizon (Median 8-15 steps)
Online-Mind2Web	√	300	136	English	WebJudge (key screenshot selection)	Avg. Step = 8.4, 25% Tasks ≥ 11 steps
Mind2Web 2	√	130	--	English	Agent-as-a-Judge	> 50 actions/task
WebRetriever	√	1550	800	Chinese + English	Three-stage protocol	Not publicly reported
BrowserBench (Ours)	√	350	254	Chinese + English	LLM-as-a-Judge	Avg. Step = 37.9, Max Step = 100

Figure 1: Three obstacles to locating the decisive step. (a) Public SFT data contains only successful trajectories. (b) Rollouts share long prefixes; live sessions cannot be restored. (c) Short benchmarks cause errors to surface immediately rather than propagating silently.

matched action prefixes. BrowserBench (Section 7) provides long-horizon, solvability-calibrated evaluation. Our contributions are:

- **We propose RUIC-SFT**, a curriculum admitting state-conditioned recovery supervision only once separable, jointly improving recovery and reducing redundant actions.
- **We introduce DAO-GRPO**, deriving unbiased step-level advantages from prefix-conditioned leave-one-out baselines, without resets or per-step model calls.
- **We construct Live-Browser-Set**, comprising 2,832 replayable trajectories with verified corrections and $\approx$7,400 grouped live-web rollouts for offline recomputation.
- **We build BrowserBench**, a bilingual long-horizon online real-web benchmark of 350 tasks over 254 sites averaging 37.9 human-demonstrated steps.

2 Related Work

2.1 Browser Agents

ReAct (Yao et al., 2022) established the reasoning-and-acting loop still used by most browser agents (Zheng et al., 2024; Lai et al., 2024; Awadallah et al., 2025; Gupta et al., 2026; Yang et al., 2026; Qin et al., 2025; Wang et al., 2025a; Liu et al., 2025a; Wang et al., 2026b; Xu et al., 2026b; Team et al., 2026b; Zhou et al., 2026). Frontier foundation models have recently advanced computer-use capabilities, but training data remains demonstration-dominated (Deng et al., 2023; Xu et al., 2025). Few public browser datasets annotate the exact divergence step of a failed trajectory or verify that a correction restores it. Methods that learn from failure (Shinn et al., 2023; Song et al., 2024; Yuan et al., 2025) also rely on resettable environments without execution-level verification. Evaluation has likewise shifted from self-hosted sandboxes (Zhou et al., 2024) to live-web suites (He et al., 2024; Xue et al., 2025; Gou et al., 2026; Lyu et al., 2025; Wei et al., 2025a), but site coverage remains narrow and the Chinese web largely unevaluated.

2.2 Agentic Reinforcement Learning

Agentic reinforcement learning is commonly studied in offline and online settings. Offline methods (Rafailov et al., 2023) avoid environment interaction, but cannot capture the evolving error distribution induced by deployment-time exploration. Online optimization is therefore better suited to long-horizon browser use, where mistakes propagate and recovery must be learned from actual

interaction. Group-relative critic-free objectives (Shao et al., 2024; Yu et al., 2026), usable in both settings, have recently become common for web agents (Qi et al., 2025; Zhang et al., 2025; Wei et al., 2025b; Wang et al., 2025c; Bai et al., 2026). Their limitation is not the optimization regime but the credit granularity: a single trajectory-level advantage is applied uniformly to all steps.

Two families resolve credit below the trajectory level. The first intervenes on the environment: VinePPO (Kazemnejad et al., 2024) re-rolls from intermediate states, while BPO (He et al., 2026a) and tree objectives (Li et al., 2025; Ji et al., 2026) fork siblings from a saved snapshot. This requires restoring a snapshot to faithfully reproduce the original page dynamics, a condition local containers satisfy but live remote services do not. The second matches steps across rollouts by observation, using identical states (GiGPO (Feng et al., 2026)), k-step context hierarchies (HGPO (He et al., 2026b)), or merged observation graphs (Cheng et al., 2026; Yan et al., 2026), which live pages rarely supply.

DAO-GRPO leaves the environment untouched, adds no per-step model call, and matches on actions rather than observations, so its prefix-sharing set stays non-degenerate even when no two page states coincide. Furthermore, observation-matching methods degrade to trajectory-level credit assignment as state recurrence thins on live pages, whereas DAO-GRPO interpolates to the group-level baseline via empirical Bayes shrinkage when its prefix-sharing set becomes a singleton.

3 Problem Setup

We model browser-agent execution as an episodic partially observable decision process. Given instruction g, the agent interacts for at most $T_{\max}$ steps; at step t it receives observation o_t and emits response y_t, from which structured action a_t is parsed, executed, and answered with feedback e_t. Actions are function calls over 33 browser primitives (Appendix B.1); a trajectory is $\tau = \{(o_t, c_t, y_t, a_t, e_t)\}_{t=1}^{T}$ with $T \le T_{\max}$. Success is judged externally; no ground-truth reward is observed during execution. We sample K trajectories independently per task.

The primary observation is a structured page representation S_t, optionally augmented by a viewport screenshot V_t. Since representations span thousands of tokens, the policy conditions on a maintained context $c_t = \mathcal{R}(g, o_t, \{a_u, e_u\}_{u<t})$, where $\mathcal{R}$ retains history and drops representations invalidated by navigation, so $c_t \not\subseteq c_{t+1}$ in general. Every step's exact context is recorded for offline rescoring.

Two constraints shape feasible credit assignment. First, live sessions cannot be faithfully restored, so intermediate-state re-rollouts are unreliable. Second, serialized observations contain dynamic noise and rarely recur across rollouts. Actions, by contrast, come from a fixed vocabulary of typed primitives, making prefix-based matching feasible. Section 6 designs DAO-GRPO around these constraints.

4 Live-Browser-Set

Because live web sessions preclude state rollbacks (Section 3), offline curation provides the sole verifiable source of step-level supervision. Live-Browser-Set supplies 2,832 trajectories (21,183 steps) across 3,500 real Chinese and English web tasks, partitioned into three functional subsets (Table 1; schema and protocol in Appendix D.1, construction in Appendix D). D_g supplies the execution prior: verified successes spanning navigation, search, multi-stage forms, cross-tab coordination and file deliverables. D_u targets 130 control instances across ten widget families (date pickers, cascaders, sliders, modals, tab groups, rich-text toolbars, etc.). D_r records, for each erroneous step t^*, the incorrect action, error observation, error category, and a correction verified by sandbox re-execution. The online part adds $1{,}300+$ goal-only tasks and $\approx 7{,}400$ grouped rollouts (up to 50 steps) with per-step contexts, canonical action signatures, key-point marks and terminal outcomes—exactly what the objective of Section 6 consumes offline.

Table 1: Composition of Live-Browser-Set.

Part	Subset	#Traj.	Steps	Supplies
Offline	D_g general	2,003	2–57	bilingual, verified
	D_u UI-spec.	412	2–21	130 controls
	D_r recovery	417	3–34	labelled error, correction
Online	task pool	—	—	1,300+ tasks
	rollouts	7,400	2–50	K=6 groups

5 RUIC-SFT: RECOVERY- AND UI-SPECIALIZED CURRICULUM SFT

The three subsets of Section 4 differ not only in content but in the type of function they specify, and this determines the order in which they can be learned (Figure 2, left).

An identifiability constraint between subsets. D_g and D_u specify unconditional mappings, whereas D_r specifies a conditional one: given an off-path context, emit a corrective action. This conditioning event is latent, so learning from D_r requires a statistic that separates off-path from on-path contexts. D_r alone cannot provide it, because its paired negatives reflect annotator-injected errors rather than the policy's own evolving off-path distribution. That statistic emerges only after fitting D_g, whose on-path prior allows deviations to become detectable. Introducing D_r too early therefore fits the marginal rather than the conditional, causing corrective actions to be emitted at a context-independent rate (*over-hedging*). The bottleneck is a missing conditioning statistic rather than conflicting gradient directions, so gradient-projection remedies (Yu et al., 2020) do not address it. D_u imposes no ordering constraint with respect to D_g.

Measuring the relation. While the separating statistic is unavailable, reducing loss on D_r shifts the unconditional action distribution toward corrective actions, increasing loss on D_g. We measure this directed effect. Let θ_λ denote parameters at normalized progress $\lambda = t/T$, let $\theta' = \theta_\lambda - \nu \nabla \mathcal{L}_A(\theta_\lambda)$ be a single probe step on subset A, and define

$$\mathcal{I}_{A\to B}(\lambda) \;=\; -\,\frac{\mathcal{L}_B^{\mathrm{val}}(\theta') - \mathcal{L}_B^{\mathrm{val}}(\theta_\lambda)}{\nu \left\|\nabla \mathcal{L}_A(\theta_\lambda)\right\|}, \tag{1}$$

where $\mathcal{L}_B^{\mathrm{val}}$ is evaluated on a held-out split of B excluded from training. Positive values indicate that a step on A improves B. Since $\mathcal{I}_{A\to B}$ and $\mathcal{I}_{B\to A}$ coincide to first order when training and validation distributions match, their persistent sign asymmetry indicates that D_r's injected-error contexts differ from the held-out off-path contexts—precisely the identifiability gap motivating the curriculum (probe details in Appendix G.1). This first-order task-affinity probe follows Fifty et al. (2021). The account above predicts a specific sign pattern: $\mathcal{I}_{D_g\to D_r}(\lambda) > 0$ for all λ; $\mathcal{I}_{D_r\to D_g}(\lambda) < 0$ until that statistic becomes available and approximately zero thereafter; $\mathcal{I}_{D_u\to D_g}(\lambda) \approx 0$ throughout. The asymmetry between the first two is the discriminating observation (Figure 5).

Thresholds informed by the measurement. We identify two thresholds from the Eq. (1) curves: λ_1, where $\mathcal{I}_{D_g\to D_g}$ has halved (diminishing marginal effect of general supervision), and λ_2, the zero-crossing of $\mathcal{I}_{D_r\to D_g}$ (recovery no longer degrades the execution prior). On the 9B backbone $\lambda_1 = 0.31$ and $\lambda_2 = 0.68$ (Figure 5a); λ_2 moves from 0.79 at 4B to 0.55 at 27B (Figure 5b). The underlying phase-transition criterion transfers across architectures: as backbone capacity increases, λ_2 shifts earlier, suggesting that larger models resolve latent error conditioning with less general-execution exposure.

Schedule and objective. Let $(x, y) \sim p_g(\lambda)D_g + p_u(\lambda)D_u + p_r(\lambda)D_r$ with $p_g + p_u + p_r = 1$, and let (p_g^*, p_u^*, p_r^*) denote the terminal mixture. The mixture follows

$$(p_g,\, p_u,\, p_r)(\lambda) = \begin{cases} \big(1 - p_u^0,\;\; p_u^0,\;\; 0\big), & \lambda \in [0,\, \lambda_1], \\ \big(1 - p_u^0 - \delta\alpha,\;\; p_u^0 + \delta\alpha,\;\; 0\big), & \lambda \in (\lambda_1,\, \lambda_2], \\ \big(p_g^*,\;\; p_u^* + p_r^*(1-\beta),\;\; p_r^*\beta\big), & \lambda \in (\lambda_2,\, 1], \end{cases} \tag{2}$$

with $\delta = 1 - p_u^0 - p_g^*$, $\alpha = (\lambda - \lambda_1)/(\lambda_2 - \lambda_1)$ and $\beta = (\lambda - \lambda_2)/(1 - \lambda_2)$. With $p_u^0 = 0.2$ and $(p_g^*, p_u^*, p_r^*) = (0.5, 0.3, 0.2)$, the quoted 5:3:2 proportion is the terminal mixture. The two ordering constants are set from preliminary experiments and confirmed by Eq. (1); the two terminal-mixture constants are design choices. The objective is standard next-token cross-entropy; the curriculum acts through the sampling distribution alone (Appendix G).

Verifying that the learned behaviour is conditional. A higher recovery rate does not by itself establish conditionality. We measure the separation

$$\Delta \;=\; \Pr[\text{corrective action} \mid \text{off-path}] \;-\; \Pr[\text{corrective action} \mid \text{on-path}] \tag{3}$$

on paired probes from D_r: for each injected error the off-path context is paired with the same step of the successful trajectory. A policy fitting the marginal exhibits small Δ; one fitting the conditional separates them. Since directly measuring Δ on live pages is impractical, we instead test whether

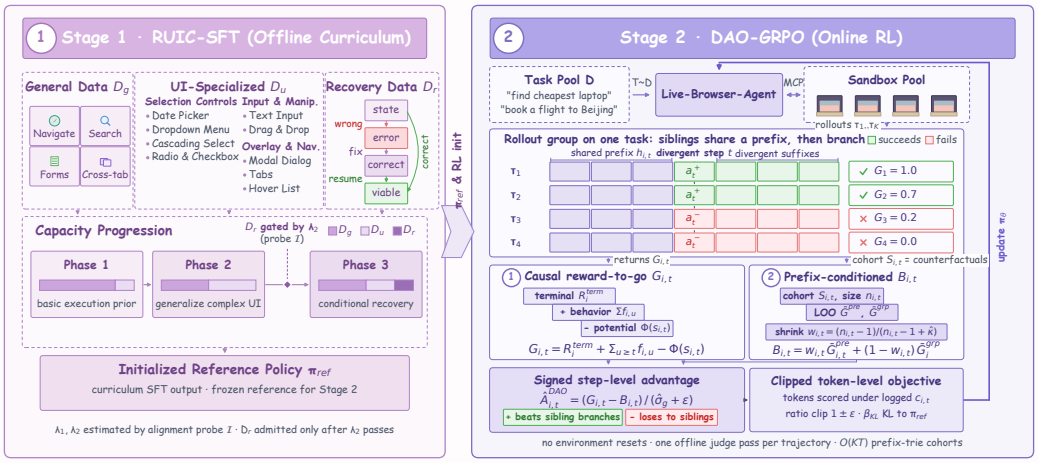


Figure 2: Two-stage training pipeline of Live-Browser-Agents.

corrective behaviour is selective: conditional recovery should maintain high recovery while reducing redundant steps (Section 8.2).

6 Divergence-Attributed Online GRPO (DAO-GRPO)

Live-web deployment imposes two constraints on step-level credit assignment: sessions cannot be rolled back (stateful backends, anti-bot systems), and serialized observations carry dynamic noise (timestamps, shifting DOM node IDs) that degenerates observation-matching methods into singletons. DAO-GRPO therefore conditions advantage estimation on canonical action-prefix cohorts: sibling rollouts sharing an action history form counterfactual branches at divergence steps, and a leave-one-out baseline over these branches yields signed step-level credit without environment resets or observation matching. The only auxiliary model is a single judge (Qwen3.7-Plus) that evaluates each completed trajectory in one offline pass—returning a terminal score in $[0, 1]$ together with per-step key-point marks—and is never queried during interaction, adding no per-step model calls.

6.1 Causal Reward-to-Go with State-Potential Baseline

For each task g, we sample K independent rollouts $\{\tau_i\}_{i=1}^{K}$. The judge's single offline pass over each completed trajectory returns (i) a terminal score $R_i^{\text{term}} \in [0, 1]$ against the task's success criterion and (ii) per-step marks $k(s_{i,t})$ over the task's K_p key points; it is the only auxiliary model in the reward pipeline. To keep early missteps visible under whole-trajectory averaging and prevent later penalties from reaching back to preceding correct actions, we define the *causal reward-to-go* $G_{i,t} = R_i^{\text{term}} + \sum_{u=t}^{T_i} f_{i,u} - \omega\, \Phi(s_{i,t})$, where $\Phi(s_{i,t}) = c_p \cdot k(s_{i,t})/K_p \in [0, c_p]$ ($c_p = 0.3$) measures judge-marked milestone progress up to step t ($\Phi(s_{i,1}) = 0$), acting as a causal, action-independent baseline that attenuates cross-trajectory progress variance ($\omega = 1$). The auxiliary behavioral term $f_{i,t} = -c_{\text{exec}} \mathbb{I}[a_{i,t} \in \mathcal{A}_{\text{invalid}}] - c_{\text{loop}} \mathbb{I}[a_{i,t} = a_{i,t-1}]$ provides sanity guardrails against degenerate exploration rather than dense reward shaping. Static coefficients (≤ 0.03, an order of magnitude below R_i^{term}) prevent computational wastage in stagnant loops across all tasks without per-benchmark tuning (Appendix G).

6.2 Prefix-Conditioned Leave-One-Out Advantage

Let $h_{i,t} = (a_{i,1}, \ldots, a_{i,t-1})$ denote the canonical action prefix of trajectory i up to step $t - 1$, where $a \simeq a'$ denotes canonical semantic equivalence over primitive types and element attributes (Appendix B.2). We define the prefix-sharing cohort:

$$S_{i,t} = \big\{\, j \in [K] : T_j \geq t,\ a_{j,u} \simeq a_{i,u}\ \forall u < t \,\big\}, \qquad n_{i,t} = |S_{i,t}|. \tag{4}$$

When $n_{i,t} \geq 2$, trajectories in $S_{i,t}$ share the same historical decisions. Rollouts branching into alternative actions at step t establish an empirical counterfactual set for evaluating the decision $a_{i,t}$.

Two-Level Leave-One-Out Baseline. We eliminate self-inclusion bias via a two-level leave-one-out (LOO) formulation (Kool et al., 2019), contrasting prefix against group baselines:

$$\bar{G}_{i,t}^{\text{pre}} = \frac{1}{n_{i,t}-1} \sum_{j \in S_{i,t} \setminus \{i\}} G_{j,t}, \qquad \bar{G}_{i}^{\text{grp}} = \frac{1}{K-1} \sum_{j \neq i} G_{j,1}. \tag{5}$$

Empirical Bayes Shrinkage. As rollouts diverge at deeper horizons, cohort size $n_{i,t}$ naturally decays. We mitigate small-sample variance via Bühlmann credibility shrinkage (Bühlmann, 1967), interpolating between prefix and group baselines:

$$B_{i,t} = w_{i,t}\, \bar{G}_{i,t}^{\text{pre}} + (1 - w_{i,t})\, \bar{G}_{i}^{\text{grp}}, \qquad w_{i,t} = \frac{\max(0, n_{i,t}-1)}{\max(0, n_{i,t}-1) + \hat{\kappa}}, \tag{6}$$

where $\hat{\kappa} = \hat{\sigma}_w^2 / \hat{\tau}^2$ is the plug-in credibility ratio (Proposition 2). When rollouts share identical paths ($n_{i,t} \gg 1$), $w_{i,t} \to 1$ isolates the branch-specific delta; when a rollout isolates ($n_{i,t} = 1$), $w_{i,t} = 0$ smoothly falls back to the group-level baseline.

Signed Step-Level Advantage. The final step advantage is normalized by group standard deviation:

$$\hat{A}_{i,t}^{\text{DAO}} = \frac{G_{i,t} - B_{i,t}}{\hat{\sigma}_g + \delta}, \qquad \hat{\sigma}_g = \text{std}\big(\{G_{j,1}\}_{j=1}^{K}\big), \tag{7}$$

where groups with $\hat{\sigma}_g = 0$ are skipped. Unlike heuristic multiplicative weighting schemes that amplify errors on successful paths, $\hat{A}_{i,t}^{\text{DAO}}$ is *intrinsically signed*: if decision $a_{i,t}$ leads to success while siblings branching from $h_{i,t}$ fail, $\hat{A}_{i,t}^{\text{DAO}} > 0$; conversely, if $a_{i,t}$ induces a redundant loop while siblings succeed, $\hat{A}_{i,t}^{\text{DAO}} < 0$.

Properties. $B_{i,t}$ is conditionally independent of $a_{i,t}$ given $c_{i,t}$, preserving gradient unbiasedness before clipping and normalization (proofs and variance-reduction conditions in Appendix H).

6.3 Optimization under Recorded Contexts

In web navigation, context updates are non-monotonic: representations invalidated by page transitions or frame alterations are pruned by the context operator $\mathcal{R}$ (Section 3), implying $c_t \not\subseteq c_{t+1}$. Re-scoring an episode in a single concatenated pass creates cross-step addressing mismatches. We therefore optimize each step strictly under its logged context $c_{i,t}$:

$$\mathcal{L}(\theta) = -\mathbb{E}_{i,t}\left[\min\left(\varrho_{i,t}\hat{A}_{i,t}^{\text{DAO}}, \text{clip}(\varrho_{i,t}, 1-\varepsilon, 1+\varepsilon)\hat{A}_{i,t}^{\text{DAO}}\right)\right] + \beta_{\text{KL}} D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}}), \tag{8}$$

where $\varrho_{i,t} = \exp\left(\frac{1}{|y_{i,t}|}\sum_{k=1}^{|y_{i,t}|}\left(\log \pi_\theta(y_{i,t,k} \mid c_{i,t}, y_{i,t,<k}) - \log \pi_{\text{old}}(y_{i,t,k} \mid c_{i,t}, y_{i,t,<k})\right)\right)$ is the length-normalized sequence importance weight, which provides a valid measure transformation on structured function calls while stabilizing gradient variance over long-token sequences; here π_{ref} denotes the frozen RUIC-SFT policy, and $\varepsilon = 0.15$. The expectation is over all valid steps in the batch. With a prefix trie over canonical signatures, cohort construction costs $O(KT)$ per group, introducing zero interactive rollout overhead.

7 BrowserBench: A Long-Horizon Real-Web Benchmark

BrowserBench (Figure 3) contains 350 real-web tasks over 254 websites, 191 Chinese and 159 English, averaging 37.9 interaction steps (min 15, max 100) across eight real-use categories. Tasks are composed from verified atomic subtasks and filtered for procedural leakage, ambiguous goals and stale entities, to reduce failures attributable to defective specifications; difficulty is calibrated by empirical solvability under three reference agents into easy/medium/hard at $30/40/30\%$. Evaluation is strictly goal-only: the agent receives the instruction with no procedural hints and succeeds only by meeting the structured success criterion within a single 100-step rollout (Pass@1). Statistics, calibration, judge design, contamination control, and a comparison with prior benchmarks (Table 7) are in Appendix F.

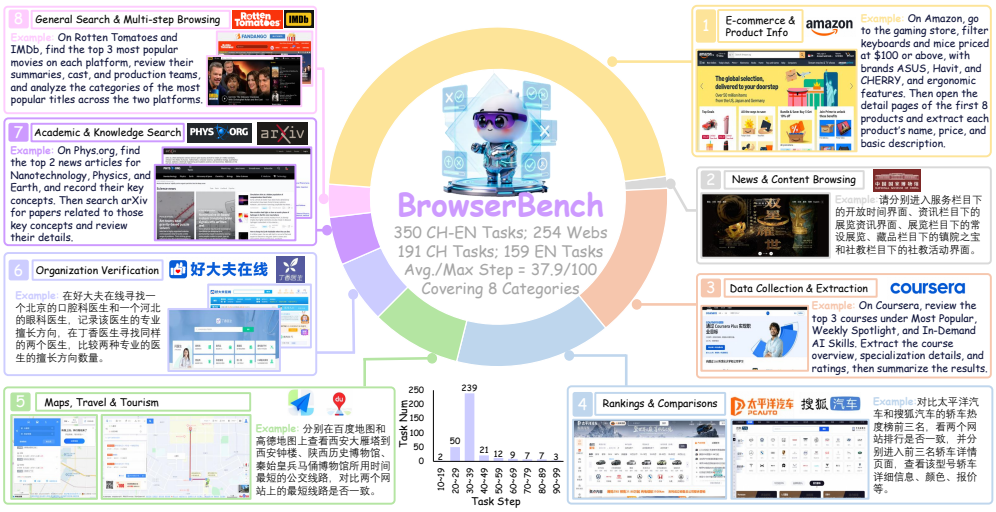


Figure 3: Overview of BrowserBench, a Chinese-English long-horizon online real-web benchmark.

8 EXPERIMENTS

Setup. We initialize from Qwen3.5-4B/9B/27B (Yang et al., 2025) and apply RUIC-SFT followed by DAO-GRPO with LoRA. The RL stage runs 150 iterations, each consuming 6 valid groups of $K=6$ after discarding zero-variance groups (26.7% of sampled groups): 5,400 rollouts are consumed by updates out of ≈7,400 generated against a cloud sandbox pool. The reward combines the terminal judge, key-point partial credit, and the behavioural terms of Section 6.1. All evaluated models, ours and baselines, interact through the same harness, tool space and observation pipeline. Judge agreement with human annotation is 96.4% on 500 audited trajectories (hyperparameters in Appendix G).

Benchmarks and Baselines. We evaluate on 3 online browser benchmarks (WebVoyager, Online-Mind2Web, BrowserBench; all Pass@1, 100-step budget) and 3 general agentic benchmarks (Tau2-Bench, BFCL-v4, Claw-Eval). Since frontier models generally outperform specialized UI agents in browser use, our browser baselines primarily consist of the latest large models, spanning closed-source (GPT-5.5 (OpenAI, 2026), Ernie-5.0 (Wang et al., 2026a), Seed2.1-Pro (ByteDance Seed, 2026)) and open-source systems (OpenWebRL-8B (Yang et al., 2026), Kimi-k3 (Team et al., 2026a), Hy4-Preview (Tencent, 2026), Qwen3.5 family (Yang et al., 2025), Qwen3.8-Max-xhigh-0902, DeepSeek-V4-Flash (Xu et al., 2026a)). For general agentic ability, we compare our 27B variant against similar-scale agents: OpenCUA-72B (Wang et al., 2026b), GUI-Owl-1.5-32B-Instruct (Xu et al., 2026b), UI-Venus-1.5-30B-A3B (Team et al., 2026b), EvoCUA-32B (Huang et al., 2026), Qwen3.5-27B (Yang et al., 2025), and Qwen-UI-Agent-27B (Zhou et al., 2026).

8.1 MAIN RESULTS

Table 2 reports browser-use success rates across three online benchmarks. Live-Browser-Agent-27B achieves the best open-source average score (82.3%), surpassing Qwen3.8-Max-xhigh-0902 (80.5%) with roughly 90× fewer total parameters and slightly edging out Kimi-k3 (82.2%). The advantage is most pronounced on WebVoyager (92.4% vs. 89.1% for Qwen3.8-Max) and Online-Mind2Web (89.3% vs. 87.3%), while BrowserBench remains the most difficult suite and Kimi-k3 retains the best open-source result there (74.5% vs. our 65.2%).

RL gains decrease with scale, but absolute performance peaks at 27B. DAO-GRPO improves the 4B, 9B, and 27B models by +9.2, +8.4, and +7.8 Avg points over RUIC-SFT. The smaller models gain more, consistent with weaker execution priors relying more on explicit step-level credit; the 27B model still reaches the highest absolute performance, showing scale and step-level RL are complementary. Step-level credit assignment proves most critical where errors are recoverable and decisive steps are sparse: at 27B, DAO-GRPO's margin over RUIC-SFT expands with horizon length (+4.4 on WebVoyager, +8.6 on Online-Mind2Web, and +10.3 on BrowserBench), as gains concentrate heavily on hard and $>$ 50-step workflows (Figure 6). The 9B model gains only +4.9 on BrowserBench, whose

Table 2: Pass@1 success rate (%) on three online browser benchmarks, all with a 100-step budget. Bold numbers are the best open-source model results on each benchmark.

Model	WebVoyager	Online-Mind2Web	BrowserBench	Avg.
Closed-source Models				
GPT-5.5	96.4	93.0	67.4	85.6
Ernie-5.0 (>1T)	58.9	39.7	24.9	41.2
Seed2.1-Pro	80.2	75.0	40.2	65.1
Open-source Models				
OpenWebRL-8B	64.4	57.7	34.3	52.1
Kimi-k3 (2.8T-A104B)	83.8	88.3	**74.5**	82.2
Hy4-Preview (770B-A49B)	83.5	71.7	56.6	70.6
Qwen3.5-4B	42.8	25.3	16.3	28.1
Qwen3.5-9B	42.0	29.0	22.9	31.3
Qwen3.5-27B	63.0	63.3	37.1	54.5
DeepSeek-V4-Flash (284B-A13B)	78.4	72.0	48.9	66.4
Qwen3.8-Max-xhigh-0902 (2.4T-A95B)	89.1	87.3	65.2	80.5
Live-Browser-Agents (Ours)				
Live-Browser-Agent-4B-RUIC-SFT	58.0	46.7	24.3	43.0
Live-Browser-Agent-4B	68.6	54.0	34.0	52.2
Live-Browser-Agent-9B-RUIC-SFT	65.5	51.0	38.0	51.5
Live-Browser-Agent-9B	76.5	60.3	42.9	59.9
Live-Browser-Agent-27B-RUIC-SFT	88.0	80.7	54.9	74.5
Live-Browser-Agent-27B	**92.4**	**89.3**	65.2	**82.3**

tasks often exceed the 50-step training horizon, suggesting rollout depth, not credit quality, is the bottleneck at smaller scale. BrowserBench is also the only suite where we trail an open-weight frontier model (Kimi-k3, 74.5 vs. 65.2), leaving bilingual long-horizon robustness as the main open challenge.

Browser-centric RL preserves general agentic capabilities. Table 3 shows that Live-Browser-Agent-27B improves over its Qwen3.5-27B initialization on all three general agentic benchmarks (67.2 → 73.8 average), indicating that browser-centric training does not erode general tool use; it also edges out the specialized Qwen-UI-Agent-27B (72.4%).

Table 3: Agentic performance: browser-grounded training preserves broader tool use.

Model	Tau2	Claw Avg3	Claw Pass[3]	BFCL-v4	Avg
OpenCUA-72B	14.4	26.4	0.5	28.3	17.4
GUI-Owl-1.5-32B-Instruct	6.1	29.6	5.5	32.7	18.5
UI-Venus-1.5-30B-A3B	22.7	30.6	5.5	19.8	19.7
EvoCUA-32B-20260105	48.9	46.3	6.5	48.8	37.6
Qwen3.5-27B	89.2	66.9	41.2	71.3	67.2
Qwen-UI-Agent-27B	89.9	73.5	51.8	74.2	72.4
Live-Browser-Agent-27B	**91.6**	**74.1**	**52.8**	**76.5**	**73.8**

8.2 Ablations

All ablations use the Qwen3.5-9B backbone evaluated on BrowserBench.

Corpus subsets and curriculum (Table 4). D_u enhances execution success rate without aiding recovery; adding D_r in a fixed uniform mixture approximately doubles recovery SR but heavily inflates step counts (23.1 → 27.6 and 29.3 steps) due to over-hedging. The full curriculum (Row 6) simultaneously optimizes success rate (38.0%), recovery rate (18.5%), and step efficiency (22.4 steps). Because success rate increases alongside reduced trajectory length, the lower step count reflects the elimination of redundant exploratory loops and defensive hedging rather than premature task abandonment.

DAO-GRPO components (Table 5). Vanilla online GRPO yields limited gains over SFT (38.0% → 39.4%). Causal behavioral feedback f via reward-to-go adds +0.9%, and potential baseline Φ contributes +0.7% by stabilizing cross-rollout progress. Crucially, prefix-conditioned leave-one-out credit delivers the sharpest jump on recovery (+2.7 points; SR +1.1), while empirical Bayes shrinkage controls small-cohort variance at deep horizons, peaking at 42.9% SR and 32.8% recovery.

Comparison against observation matching. GiGPO and HGPO only marginally beat trajectory-level GRPO: dynamic DOM mutations and timestamps shatter state recurrence, with over 84% of

Table 4: Ablation on RUIC-SFT subsets.

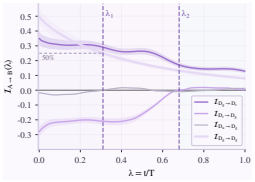
Variant	SR	Recov.	Steps
(1) Base SFT (D_g only)	32.0	8.5	24.8
(2) $+D_u$, fixed mixture	34.9	9.0	23.1
(3) $+D_r$, fixed mixture	35.5	16.4	27.6
(4) $+D_u+D_r$, fixed mixture	36.6	17.1	29.3
(5) $+D_u+D_r$, early $D_r^\dagger$	35.1	16.6	30.5
(6) **RUIC-SFT (curriculum)**	**38.0**	**18.5**	**22.4**

$^\dagger D_r$ enters at 40% from step 0, annealed to 20%.

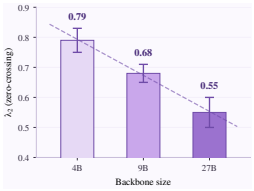
Table 5: Ablation on DAO-GRPO components.

Variant	RTG	Φ	LOO	EB	SR	Recov.
(1) Vanilla online GRPO	×	×	×	×	39.4	24.8
(2) +Causal RTG (f)	✓	×	×	×	40.3	26.1
(3) +Potential (Φ)	✓	✓	×	×	41.0	27.5
(4) +Prefix LOO	✓	✓	✓	×	42.1	30.2
(5) **DAO-GRPO (Ours)**	✓	✓	✓	✓	**42.9**	**32.8**
GiGPO‡	×	×	×	×	39.9	25.4
HGPO‡	×	×	×	×	40.4	26.6

RTG: causal f. Φ: potential baseline. **LOO**: prefix leave-one-out. **EB**: empirical Bayes shrinkage. ‡Observation-matching baselines.

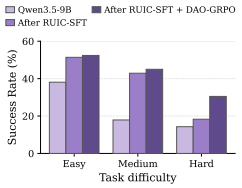


(a) Directed alignment

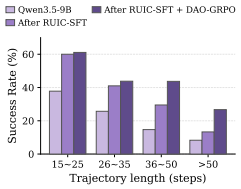


(b) λ_2 vs. backbone size

Figure 5: Directed alignment (Eq. 1): (a) $D_g \to D_r$ stays positive while $D_r \to D_g$ is negative until λ_2; (b) λ_2 decreases with backbone size.

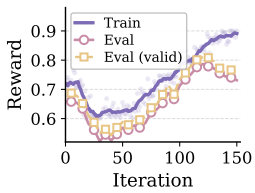


(a) SR by task difficulty

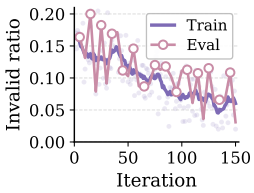


(b) SR by trajectory length

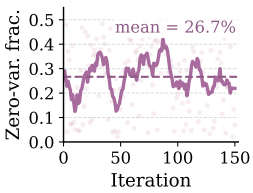
Figure 6: Success rate by difficulty and horizon. DAO-GRPO gains grow with both task difficulty and trajectory length.



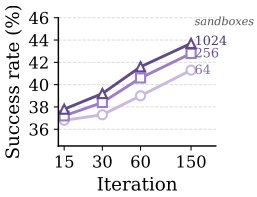
(a) Reward



(b) Invalid actions



(c) Zero-variance groups



(d) Rollout scaling

Figure 7: Online RL training dynamics of Live-Browser-Agent-9B.

observation clusters collapsing to singletons. DAO-GRPO instead matches on canonical action signatures, keeping cohorts non-degenerate ($n_{i,t} \geq 2$) at critical branching points.

8.3 Performance Dissection by Task Difficulty and Horizon

Figure 6 decomposes 9B performance across difficulty and horizon. DAO-GRPO gains concentrate heavily on hard tasks ($18.1 \to 30.5$) and trajectories beyond 50 steps ($13.3 \to 26.7$), validating that localized credit matters most where terminal feedback becomes sparse. In contrast, RUIC-SFT lifts easy and medium tasks uniformly ($+3.8$ on hard). Training dynamics (Figure 7) confirm steady convergence: invalid actions decrease from 15.1% to 6.1%, while uniform zero-variance groups (26.7%) are filtered online to maintain informative gradient signals (Appendix K).

9 Conclusion

We present a two-stage framework for long-horizon web agents: RUIC-SFT gates recovery supervision until error states become separable, and DAO-GRPO attributes step-level credit via prefix-conditioned leave-one-out baselines without environment resets. Across online browser benchmarks, our 27B model matches or surpasses proprietary frontier models while preserving general agentic capabilities. Evaluations on our bilingual BrowserBench validate that DAO-GRPO gains scale with horizon length.

AI USE STATEMENT

Large language models were used during the preparation of this manuscript solely to assist with language polishing, proofreading, and routine code prototyping. All scientific ideas, experimental designs, empirical analyses, and conclusions were independently conceived, conducted, and verified by the authors. No AI system was used to generate or manipulate experimental data.

ETHICS STATEMENT

All web trajectories were collected from publicly accessible pages under the terms of service of each site. The action space deliberately excludes credential entry, payment confirmation, and any operation that modifies persistent user data. The `solve_captcha` primitive is registered only in controlled testing environments with explicit authorization from site operators, and is disabled during all reported evaluations. Person-identifiable information encountered during browsing was filtered by an automated PII detector before corpus release. Human annotators were compensated above local minimum wage and gave informed consent.

REFERENCES

Ahmed Awadallah, Yash Lara, Raghav Magazine, Hussein Mozannar, Akshay Nambi, Yash Pandya, Aravind Rajeswaran, Corby Rosset, Alexey Taymanov, Vibhav Vineet, et al. Fara-7b: An efficient agentic model for computer use. *arXiv preprint arXiv:2511.19663*, 2025.

Hao Bai, Alexey Taymanov, Tong Zhang, Aviral Kumar, and Spencer Whitehead. Webgym: Scaling training environments for visual web agents with realistic tasks. *arXiv preprint arXiv:2601.02439*, 2026.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Hans Bühlmann. Experience rating and credibility. *ASTIN Bulletin: The Journal of the IAA*, 4(3): 199–207, 1967.

ByteDance Seed. Seed2.0. `https://seed.bytedance.com/en/seed2`, 2026. Released: 2026-02-14. Accessed: 2026-07-19.

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.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. *Advances in Neural Information Processing Systems*, 36:28091–28114, 2023.

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.

Chris Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. Efficiently identifying task groupings for multi-task learning. *Advances in Neural Information Processing Systems*, 34: 27503–27516, 2021.

Boyu Gou, Zanming Huang, Yuting Ning, Yu Gu, Michael Lin, Weijian Qi, Andrei Kopanev, Botao Yu, Bernal Jimenez Gutierrez, Yiheng Shu, et al. Mind2web 2: Evaluating agentic search with agent-as-a-judge. *Advances in Neural Information Processing Systems*, 38, 2026.

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

Tanmay Gupta, Piper Wolters, Zixian Ma, Peter Sushko, Rock Yuren Pang, Diego Llanes, Yue Yang, Taira Anderson, Boyuan Zheng, Zhongzheng Ren, et al. Molmoweb: Open visual web agent and open data for the open web. *arXiv preprint arXiv:2604.08516*, 2026.

Bowei He, Yankai Chen, Xiaokun Zhang, and Xue Liu. Branching policy optimization: Sandbox-native language agent reinforcement learning. *arXiv preprint arXiv:2607.14171*, 2026a.

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. Webvoyager: Building an end-to-end web agent with large multimodal models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6864–6890, 2024.

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Hongming Zhang, Tianqing Fang, Zhenzhong Lan, and Dong Yu. Openwebvoyager: Building multimodal web agents via iterative real-world exploration, feedback and optimization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27545–27564, 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, 2026b.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

Mianqiu Huang, Taofeng Xue, Chong Peng, Jinrui Ding, Sicheng Fan, Jiale Hong, Yufei Gao, Xiaocheng Zhang, Linsen Guo, Xin Yang, et al. Evocua-1.5: Online reinforcement learning for multi-turn computer-use agents. *arXiv preprint arXiv:2607.09773*, 2026.

Yuxiang Ji, Ziyu Ma, Yong Wang, Guanhua Chen, Xiangxiang Chu, and Liaoni Wu. Tree search for llm agent reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 87362–87388, 2026.

Amirhossein Kazemnejad, Milad Aghajohari, Eva Portelance, Alessandro Sordoni, Siva Reddy, Aaron Courville, and Nicolas Le Roux. Vineppo: Refining credit assignment in rl training of llms. *arXiv preprint arXiv:2410.01679*, 2024.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Russ Salakhutdinov, and Daniel Fried. Visualwebarena: Evaluating multimodal agents on realistic visual web tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 881–905, 2024.

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

Hanyu Lai, Xiao Liu, Iat Long Iong, Shuntian Yao, Yuxuan Chen, Pengbo Shen, Hao Yu, Hanchen Zhang, Xiaohan Zhang, Yuxiao Dong, et al. Autowebglm: A large language model-based web navigating agent. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 5295–5306, 2024.

Yizhi Li, Qingshui Gu, Zhoufutu Wen, Ziniu Li, Tianshun Xing, Shuyue Guo, Tianyu Zheng, Xin Zhou, Xingwei Qu, Wangchunshu Zhou, et al. Treepo: Bridging the gap of policy optimization and efficacy and inference efficiency with heuristic tree-based modeling. *arXiv preprint arXiv:2508.17445*, 2025.

Zhaoyang Liu, JingJing Xie, Zichen Ding, Zehao Li, Bowen Yang, Zhenyu Wu, Xuehui Wang, Qiushi Sun, Shi Liu, Weiyun Wang, et al. Scalecua: Scaling open-source computer use agents with cross-platform data. *arXiv preprint arXiv:2509.15221*, 2025a.

Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual-rft: Visual reinforcement fine-tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2034–2044, 2025b.

Zhengxi Lu, Yuxiang Chai, Yaxuan Guo, Xi Yin, Liang Liu, Hao Wang, Han Xiao, Shuai Ren, Pengxiang Zhao, Guangyi Liu, et al. Ui-r1: Enhancing efficient action prediction of gui agents by reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 17608–17616, 2026.

Yougang Lyu, Xiaoyu Zhang, Lingyong Yan, Maarten de Rijke, Zhaochun Ren, and Xiuying Chen. Deepshop: A benchmark for deep research shopping agents. *arXiv preprint arXiv:2506.02839*, 2025.

Andrew Y Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, volume 99, pp. 278–287. Citeseer, 1999.

OpenAI. GPT-5.5 System Card. `https://deploymentsafety.openai.com/gpt-5-5`, 2026. April 2026. Accessed: 2026-07-19.

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

Yujia Qin, Yining Ye, Junjie Fang, Haoming Wang, Shihao Liang, Shizuo Tian, Junda Zhang, Jiahao Li, Yunxin Li, Shijue Huang, et al. Ui-tars: Pioneering automated gui interaction with native agents. *arXiv preprint arXiv:2501.12326*, 2025.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

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.

Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, et al. Vlm-r1: A stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*, 2025.

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.

Yifan Song, Da Yin, Xiang Yue, Jie Huang, Sujian Li, and Bill Yuchen Lin. Trial and error: Exploration-based trajectory optimization of llm agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7584–7600, 2024.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y Charles, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026a.

Venus Team, Changlong Gao, Zhangxuan Gu, Yulin Liu, Xinyu Qiu, Shuheng Shen, Yue Wen, Tianyu Xia, Zhenyu Xu, Zhengwen Zeng, et al. Ui-venus-1.5 technical report. *arXiv preprint arXiv:2602.09082*, 2026b.

Tencent. Hy4-preview model card. `https://hf-mirror.com/tencent/Hy4-preview`, 2026. Hugging Face mirror, accessed: 2026-09-21.

Haifeng Wang, Hua Wu, Tian Wu, Yu Sun, Jing Liu, Dianhai Yu, Yanjun Ma, Jingzhou He, Zhongjun He, Dou Hong, et al. Ernie 5.0 technical report. *arXiv preprint arXiv:2602.04705*, 2026a.

Haoming Wang, Haoyang Zou, Huatong Song, Jiazhan Feng, Junjie Fang, Junting Lu, Longxiang Liu, Qinyu Luo, Shihao Liang, Shijue Huang, et al. Ui-tars-2 technical report: Advancing gui agent with multi-turn reinforcement learning. *arXiv preprint arXiv:2509.02544*, 2025a.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025b.

Xinyuan Wang, Bowen Wang, Dunjie Lu, Junlin Yang, Tianbao Xie, Junli Wang, Jiaqi Deng, Xiaole Guo, Yiheng Xu, Chen Wu, et al. Opencua: Open foundations for computer-use agents. *Advances in Neural Information Processing Systems*, 38:139756–139806, 2026b.

Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, et al. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*, 2025c.

Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford, Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese. Browsecomp: A simple yet challenging benchmark for browsing agents. *arXiv preprint arXiv:2504.12516*, 2025a.

Zhepei Wei, Wenlin Yao, Yao Liu, Weizhi Zhang, Qin Lu, Liang Qiu, Changlong Yu, Puyang Xu, Chao Zhang, Bing Yin, et al. Webagent-r1: Training web agents via end-to-end multi-turn reinforcement learning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 7920–7939, 2025b.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026a.

Haiyang Xu, Xi Zhang, Haowei Liu, Junyang Wang, Zhaozai Zhu, Shengjie Zhou, Xuhao Hu, Feiyu Gao, Junjie Cao, Zihua Wang, et al. Mobile-agent-v3. 5: Multi-platform fundamental gui agents. *arXiv preprint arXiv:2602.16855*, 2026b.

Yiheng Xu, Dunjie Lu, Zhennan Shen, Junli Wang, Zekun Wang, Yuchen Mao, Caiming Xiong, and Tao Yu. Agenttrek: Agent trajectory synthesis via guiding replay with web tutorials. In *International Conference on Learning Representations*, volume 2025, pp. 79822–79843, 2025.

Tianci Xue, Weijian Qi, Tianneng Shi, Chan Hee Song, Boyu Gou, Dawn Song, Huan Sun, and Yu Su. An illusion of progress? assessing the current state of web agents. *arXiv preprint arXiv:2504.01382*, 2025.

Hongxi Yan, Ziyue Huang, Shichao Fan, and Qingjie Liu. Gated-bepo: Confidence-gated bellman credit assignment for large language model agents. *arXiv preprint arXiv:2608.06861*, 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.

Rui Yang, Qianhui Wu, Yuxi Chen, Hao Bai, Wenlin Yao, Hao Cheng, Baolin Peng, Huan Zhang, Tong Zhang, and Jianfeng Gao. Openwebrl: Demystifying online multi-turn reinforcement learning for visual web agents. *arXiv preprint arXiv:2606.02031*, 2026.

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*, 2022.

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, 2026.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33: 5824–5836, 2020.

Siyu Yuan, Zehui Chen, Zhiheng Xi, Junjie Ye, Zhengyin Du, and Jiecao Chen. Agent-r: Training language model agents to reflect via iterative self-training. *arXiv preprint arXiv:2501.11425*, 2025.

Hanchen Zhang, Xiao Liu, Bowen Lv, Xueqiao Sun, Bohao Jing, Iat Long Iong, Zhenyu Hou, Zehan Qi, Hanyu Lai, Yifan Xu, et al. Agentrl: Scaling agentic reinforcement learning with a multi-turn, multi-task framework. *arXiv preprint arXiv:2510.04206*, 2025.

Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. Gpt-4v (ision) is a generalist web agent, if grounded. *arXiv preprint arXiv:2401.01614*, 2024.

Hanzhang Zhou, Panrong Tong, Xu Zhang, Quyu Kong, Chenglin Cai, Tianyu Xia, Gongjie Zhang, Jianan Zhang, Long Li, Long Chen, et al. Qwen-ui-agent technical report: Toward next-generation real-world centric foundation gui agents. *arXiv preprint arXiv:2607.28227*, 2026.

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, et al. Webarena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations*, volume 2024, pp. 15585–15606, 2024.

Yifei Zhou, Qianlan Yang, Kaixiang Lin, Min Bai, Xiong Zhou, Yu-Xiong Wang, Sergey Levine, and Li Erran Li. Proposer-agent-evaluator (pae): Autonomous skill discovery for foundation model internet agents. In *Forty-second International Conference on Machine Learning*, 2025.

A Appendix

Appendix Overview

B Browser harness layer: action space, schema-validated execution, and the context reconstruction operator $\mathcal{R}$.

C Extended related work.

D Live-Browser-Set details: record schema, per-subset construction pipelines, quality control, release format and intended use.

E The self-reinforcing data pipeline that refreshes the corpus as websites change.

F BrowserBench details: statistics, difficulty calibration, judge design.

G Implementation details and hyperparameters.

H Proofs of theoretical results.

I Baseline citations.

J Reliability of the progress estimator Φ.

K Additional results: rollout-group case studies, qualitative trajectory.

L Limitations.

B Browser Harness Layer

Long-horizon browser interaction requires more than a strong policy: model outputs must be parsed, validated and executed reliably against live pages, and the context fed back to the model must stay faithful to the current task state. We therefore build a *browser harness* above the policy. The harness is the execution substrate shared by Live-Browser-Set construction, RUIC-SFT replay, DAO-GRPO rollouts, and all evaluation (including baseline evaluation), which is what makes cross-stage transfer and fair comparison possible. It is deliberately presented as an engineering prerequisite rather than a contribution.

B.1 Structured tool space

The harness exposes 33 browser primitives in five functional categories (Table 6): *navigation* (URL opening, search, back, current-URL query, tab management), *interaction* (click, hover, text input, scroll, scroll-to-text, key events, dropdown selection, drag, slider, file upload), *extraction and observation* (DOM snapshot, structured and DOM-based extraction, screenshots, PDF saving, JavaScript evaluation, email retrieval), *file operations* (local read/write/replace as intermediate memory and deliverables), and *flow control* (wait, plan update, captcha solving, completion signaling). At each step the model emits one function call over this space as its response y_t. To reduce observation noise and token cost, an optimized DOM simplification pipeline preserves decision-relevant interactive elements instead of serializing raw DOM trees.

B.2 Canonical action equivalence

Two actions a and a' drawn from the 33-primitive tool space are *canonically equivalent*, written $a \simeq a'$, when they invoke the same primitive with semantically identical parameters after normalization. Normalization collapses surface variation that does not affect execution outcomes: whitespace and casing differences in text arguments, reordering of independent JSON keys, and default-valued optional parameters versus their explicit equivalents. Index-based targets are canonicalized at execution time into stable element descriptors (semantic role, accessible name, and an XPath summary) recorded in the log, so equivalence is decided from these recorded signatures—without re-executing actions, and without requiring any two page observations to be equal. This decidability is what makes prefix-trie construction in DAO-GRPO an offline string operation over logged action sequences rather than an environment query.

Table 6: Browser agent action space (33 primitives). ✓ marks required parameters. Index-based actions operate on DOM element identifiers extracted from the current page state; `solve_captcha` and `get_recent_emails` are registered only when the corresponding callback/integration is enabled.

Category	Action	Key parameters (✓ = required)
Navigation	`go_to_url`	url✓, new_tab
	`go_back`	–
	`search_bing`	query✓
	`get_current_url`	–
	`switch_tab`	tab_id✓
	`close_tab`	tab_id✓
	`list_tabs`	–
	`close`	–
Interaction	`click_element_by_index`	index✓, button, while_holding_ctrl
	`hover_element_by_index`	index✓
	`input_text`	index✓, text✓, clear_existing, is_date_time_input, keep_focus
	`scroll`	direction✓, num_pages✓, frame_element_index✓
	`scroll_to_text`	text✓, frame_element_index✓
	`send_keys`	keys✓
	`get_dropdown_options`	index✓
	`select_dropdown_option`	index✓, text✓
	`drag_element`	source_index✓, target_index✓
	`set_slider_value`	index✓, value✓
	`upload_file_to_element`	index✓, path✓
Extraction & observation	`snapshot`	–
	`extract_structured_data`	query✓, extract_links✓
	`dom_extract`	instruction✓, data_schema, use_vision
	`take_screenshot`	file_name✓, full_page
	`save_page_as_pdf`	file_name✓
	`eval_javascript`	goal✓
	`get_recent_emails`	keyword, max_results
File operations	`read_file`	file_name✓
	`write_file`	file_name✓, goal✓, append, trailing_newline, leading_newline
	`replace_file_str`	file_name✓, old_str✓, new_str✓
Flow control	`wait`	seconds (default 3)
	`update_plan`	text✓
	`solve_captcha`	captcha_type, timeout
	`done`	success✓, files_to_display

B.3 Schema-validated execution and typed feedback

The harness parses each emitted call, validates its schema and parameters against the tool definitions, dispatches it to the corresponding executor in the sandbox, and returns structured feedback e_t together with the updated observation o_{t+1}. Invalid calls are not silently dropped: they are rejected with typed error messages that re-enter the context as feedback, so the policy can detect and correct its own malformed outputs. This closes the perception–decision–execution–update loop at every step and is also what makes the invalid-action-ratio metric measurable.

B.4 Task-level state management: implementing $\mathcal{R}$

Rather than concatenating all past observations into an append-only transcript, the harness maintains a decision-oriented execution context. Concretely, it (i) synchronizes the current structured state S_t; (ii) retains relevant action/feedback history; (iii) removes obsolete state blocks after navigation, since a stale DOM snapshot is actively misleading; (iv) appends compact diffs $D_{u\to v}$ for local page changes instead of a full re-serialization; and (v) attaches a screenshot V_t only when visual grounding is required. Together these rules constitute the reconstruction operator $\mathcal{R}$ defined in Section 3. The harness additionally manages intermediate artifacts (extracted content, temporary files), enabling cross-step memory and file-valued deliverables. Because the same rules are applied when replaying a logged trajectory, $\log \pi_\theta(y_{i,t} \mid c_{i,t})$ in Eq. (8) is computed on exactly the information set the policy had online.

B.5 SANDBOX ENVIRONMENT

All interactions run in a commercial cloud sandbox service providing isolated browser environments. Each task receives an independent sandbox in a standardized Linux image, ensuring consistent parallel execution, and sandboxes are recycled after each episode to guarantee state isolation. The agent communicates over a lightweight Model Context Protocol interface exposing primitives for session initialization, state acquisition, action execution, metadata retrieval, and structured output control. Action execution and state-transition retrieval are coupled in a single environment call, avoiding repeated transmission of full browser states; without this, environment communication becomes the bottleneck for trajectories with dozens of steps.

C EXTENDED RELATED WORK

C.1 BROWSER AGENTS

Progress in browser agents has been driven by stronger foundation models (Yang et al., 2025; Xu et al., 2026a; Bai et al., 2025; Wang et al., 2025b), grounding strategies, and web-specific post-training. ReAct (Yao et al., 2022) established the reasoning-and-acting paradigm; SeeAct (Zheng et al., 2024) showed GPT-4V can act as a generalist web agent with structured grounding; AutoWebGLM (Lai et al., 2024) bootstrapped a navigation agent through automated collection and RL. UI-TARS (Qin et al., 2025) and UI-TARS-2 (Wang et al., 2025a) train native GUI agents that perceive screenshots directly, while FARA (Awadallah et al., 2025), MolmoWeb (Gupta et al., 2026) and ScaleCUA (Liu et al., 2025a) show compact models with efficient agentic design or curated demonstrations can be strong. OpenWebVoyager (He et al., 2025) explores iterative real-world exploration and feedback. Evaluation broadened from self-hosted environments (Zhou et al., 2024) and static demonstrations (Deng et al., 2023) to live web benchmarks (He et al., 2024; Xue et al., 2025; Koh et al., 2024; Lyu et al., 2025). Most of these systems study architecture or training algorithm in isolation and do not jointly release recovery-oriented data, a long-horizon online objective, and bilingual long-horizon evaluation.

C.2 COMPUTER-USE AND GENERALIST GUI AGENTS

A parallel line trains computer-use agents (CUAs) for general GUI control across desktop, web and mobile; several serve as our open-source baselines. OpenCUA (Wang et al., 2026b) scales OS-level interaction supervision; GUI-Owl-1.5 (Xu et al., 2026b) builds unified cross-platform vision-language agents; UI-Venus (Team et al., 2026b) pursues compact mixture-of-experts GUI agents; EvoCUA (Huang et al., 2026) shows multi-turn online RL strengthens general tool use; Qwen-UI-Agent (Zhou et al., 2026) provides a general UI-grounded agent; and OpenWebRL (Yang et al., 2026) shows compact open models become competitive through online multimodal RL on live pages. These systems are optimized largely for atomic GUI control and evaluated under relatively short-horizon or single-platform settings; none targets recovery supervision, branch-sensitive credit under sparse terminal rewards, and bilingual long-horizon evaluation together.

C.3 AGENTIC REINFORCEMENT LEARNING

Outcome-based RL for reasoning (Guo et al., 2025) accelerated RL for interactive agents. GRPO (Shao et al., 2024) and DAPO (Yu et al., 2026) enable critic-free optimization via group-relative advantages; Visual-RFT (Liu et al., 2025b), VLM-R1 (Shen et al., 2025) and UI-R1 (Lu et al., 2026) extend RL fine-tuning to multimodal and GUI settings. For browser agents, static supervision is insufficient because tasks are interactive, stateful and path-dependent, so exploration, recovery and adaptation to unseen page variants cannot be learned from fixed demonstrations. WebRL (Qi et al., 2025) studies curriculum online optimization in WebArena; AgentRL (Zhang et al., 2025), WebAgent-R1 (Wei et al., 2025b) and RAGEN (Wang et al., 2025c) explore multi-turn GRPO-style training and self-evolution; PAE (Zhou et al., 2025) studies autonomous skill discovery; WebGym (Bai et al., 2026) provides scalable open-web environments. Two open difficulties motivate DAO-GRPO: long-horizon credit assignment under sparse rewards, addressed in prior work by dense shaping (Ng et al., 1999) or costly process reward models; and the non-append-only nature of browser contexts, which breaks the assumption underlying sequence-level objectives.

C.4 BENCHMARKS

See Table 7. WebArena (Zhou et al., 2024) offers deterministic self-hosted verification; Mind2Web (Deng et al., 2023) large offline demonstrations over static HTML; WebVoyager (He et al., 2024) live sites with LLM judgment; Online-Mind2Web (Xue et al., 2025) highlights temporal website drift; Mind2Web 2 (Gou et al., 2026), DeepShop (Lyu et al., 2025) and BrowseComp (Wei et al., 2025a) target information gathering, shopping and hard retrieval. Long horizons remain underrepresented, English dominates, and site coverage is often narrow.

D LIVE-BROWSER-SET DETAILS

D.1 RECORD SCHEMA AND REPLAYABILITY

Every trajectory is stored as JSONL with one record per step containing: task id and instruction; language, domain, and capability tags; step index; the reconstructed decision context c_t serialized into its constituent blocks (system prompt, task, retained (a_u, e_u) pairs, current S_t or diff $D_{u\to v}$, optional screenshot reference); the model response y_t; the parsed structured action a_t; typed environment feedback e_t; and any artifact produced. Because $\mathcal{R}$ (Appendix B) is deterministic given the log, a record set can be replayed to recover the exact per-step inputs used for training, which is what allows Eq. (8) to be recomputed offline by third parties.

D.2 D_g: GENERAL WEB TASKS

D_g trajectories are collected on real Chinese and English sites, retained only when the terminal state satisfies the task specification under the same structured criterion used for evaluation, and de-duplicated by (site, task template) pairs. Coverage spans navigation and search, filtering and sorting, cross-page and cross-site comparison, structured extraction into schemas, multi-stage form filling with dependent controls, cross-tab coordination, and tasks whose deliverable is a written file. We explicitly exclude BrowserBench sites and tasks to avoid contamination (Appendix F.4).

D.3 D_u: SPECIALIZED UI COMPONENT PIPELINE

Construction proceeds in four stages. **(1) Scene collection.** Automated DOM-structure scanning plus manual verification establishes a mapping among component type, website, page, and concrete control instance, giving 130 verified instances over ten widget families: date picker, dropdown menu, cascading selector, radio/checkbox group, text input/search box, drag-and-drop slider, modal dialog, tab component, hover-triggered list, and rich-text editor. **(2) Parameterized task design.** For each control type we author tasks of increasing difficulty, from single-step selection, through compound operations (e.g. range selection across months, three-level cascader with linked confirmation), to edge cases with invalid inputs or exceptional control states. **(3) Human demonstration.** Annotators operate the real controls, recording the action sequence, key operation nodes, and control-state transitions such as panel expansion, highlight change, and linked update. **(4) Augmentation.** We transfer interaction strategies across websites implementing the same widget family, generate equivalent action-sequence variants, and inject anomalous states (loading delay, disabled control, validation failure). The last category intentionally produces situations whose correct handling requires recovery, forming a bridge to D_r.

D.4 D_r: REFLECTION DATASET DETAILS

Error taxonomy. *Navigation error* (clicking a wrong link or entering a wrong page); *input error* (malformed or mis-targeted content); *sequence error* (executing steps in an invalid order, e.g. submitting before a dependent field is set); *state misinterpretation* (treating a transient or stale state as final); *loop error* (repeating an ineffective action without adaptation).

Collection. (i) **Natural error collection** extracts error segments from failed rollouts produced during self-bootstrapping; a step-level judge localizes t^* and annotators write the reflection and correction. (ii) **Manual error injection** replaces one correct action in a successful trajectory with a plausible templated mistake and constructs the corresponding awareness–reflection–correction

continuation, guaranteeing coverage of rare categories. (iii) **Contrastive pair construction** builds matched error/success trajectories from the same task and initial state, enabling preference-based training such as DPO (Rafailov et al., 2023).

Quality gates. Reflections must attribute failure to concrete actions and state transitions rather than generic description; every correction sequence is re-executed in the sandbox and discarded unless it verifiably restores a viable path; and the error-category histogram is monitored so that no single failure mode dominates.

D.5 Online part: task pool and rollout annotations

The task pool contains more than $1{,}300$ goal-only live-web tasks with domain, language, and capability tags, plus an estimated horizon used for curriculum ordering; about 83% are completable within 50 steps. The release covers all $\approx 7{,}400$ generated rollouts ($5{,}400$ consumed by updates). Each released rollout group contains $K = 6$ on-policy trajectories with the terminal judge label, the per-state potential $\Phi(s_{i,t})$, the resulting shaped increments, and per-step canonical action signatures enabling offline recomputation of the step-level advantage (Eq. 7). Groups whose returns are identical (zero variance) are excluded from the update (Section 6.1) but retained in the release with a flag, because their prevalence signals task solvability (Appendix K).

D.6 Release format, licensing and intended use

We release JSONL trajectory records, the task pool, and replay tooling. Page content is stored as simplified structured representations plus URLs and timestamps rather than verbatim site mirrors; screenshots are released only where site terms permit and are PII-filtered. The corpus is intended for research on long-horizon agent training and evaluation, not for building scraping or automated-transaction services; the action space excludes credential entry and payment confirmation.

E The Self-Reinforcing Data Pipeline

Live websites change continuously, so any fixed demonstration corpus depreciates. Live-Browser-Set is therefore maintained by a closed loop that converts ongoing interaction, including failed interaction, into fresh verified supervision. We describe it here because it explains how the corpus is produced and kept current.

(1) Unified trajectory collection. Because corpus construction, online rollouts and evaluation all run on the same harness, every executed episode (whether from DAO-GRPO training, benchmark evaluation, or deployment) is logged in an identical, training-ready format with structured actions, typed feedback, and reconstructed contexts.

(2) Automated triage. A step-level judge first separates genuine model failures from environment-induced ones (transient pages, unreachable sites), routing the latter to task-pool maintenance rather than training. It then classifies the remainder: successful episodes become candidate D_g demonstrations; failures with a localizable erroneous step become D_r candidates; failures concentrated on specific widgets are routed to the D_u pipeline as new component scenes. Training signal is thus recovered from failures rather than discarded.

(3) Verification-gated augmentation. Candidate corrections are re-executed in the sandbox and only strategies that verifiably restore a viable path enter D_r. This gate is what prevents the loop from amplifying its own mistakes, the standard failure mode of unverified self-training.

(4) Diagnostic targeting. Error-category statistics from triage and tag-wise BrowserBench results are aggregated into prioritized capability targets, which steer both task synthesis for the online pool and collection for the next SFT round.

The loop is automated where verification is cheap and human where judgment is scarce: logging, triage and correction re-execution run unsupervised, while humans author reflection reasoning, annotate complex-UI interactions, and audit borderline judge decisions. A single fixed-version judge (Qwen3.7-Plus) scores trajectories against per-task structured criteria for both benchmark evaluation and RL reward computation, so training and evaluation share one calibrated signal.

Table 7: Comparison of BrowserBench with existing web-agent benchmarks.

Benchmark	Tasks	Sites	Lang.	Online	Avg. Step	Difficulty
WebArena	812	5	EN	×	7.2	×
Mind2Web	2,350	137	EN	×	7.3	×
WebVoyager	643	15	EN	✓	15†	×
Online-Mind2Web	300	136	EN	✓	8.4	×
Mind2Web 2	130	–	EN	✓	>50	×
BrowserBench (Ours)	**350**	**254**	**CN+EN**	✓	**37.9**	✓

†Evaluation step budget; no human reference trajectory reported.
Avg. Step = human reference length where available.

In this work we instantiate the bootstrapping cycle: an initial model trained on general demonstrations only (Base SFT, Table 4) bootstraps roughly 3,000 rollouts, from which about 420 verified reflection examples and 130 UI-component scenes are distilled into the released mixture; diagnostic slices from an early BrowserBench snapshot guided the composition of the online task pool. Because triage and re-execution are automated, each additional cycle costs sandbox compute and lightweight auditing rather than proportional annotation effort. We regard sustained multi-cycle operation, including preference training over the accumulated contrastive pairs, as the primary mechanism for keeping the corpus aligned with the evolving web.

F BrowserBench Details

F.1 Statistics

Figure 3 (main text) gives the per-category and per-language distribution. Length is heavily concentrated in the 30–39 step band (239 tasks) with a long tail to 100 steps; Chinese tasks average 41.1 steps versus 34.1 for English.

F.2 Difficulty calibration

The three calibration agents are chosen from distinct capability tiers and deliberately exclude the model family we train on, so that difficulty labels do not encode our own model's idiosyncrasies. Each calibrator runs the goal-only protocol once under the 100-step budget. Boundary cases, notably tasks whose single success may be attributable to site nondeterminism, are re-run and human-reviewed. We verify (i) the easy/medium/hard split is approximately balanced across Chinese and English subsets, and (ii) mean completion length increases monotonically from easy to hard, so that "hard" is not merely a proxy for "long" or "Chinese".

F.3 Success criteria and judging

Each task carries a structured success criterion enumerating the required terminal evidence (which entities, which fields, which comparisons) rather than a required action sequence, so that alternative valid execution paths are not penalized. Judging is performed by Qwen3.7-Plus (fixed version, the same judge used for RL rewards), inspecting the terminal state and interaction log against this criterion. For the audit reported in §8 we sampled 500 trajectories across models and had them independently annotated; agreement was 96.4%, with disagreements concentrated on partially complete multi-entity extraction rather than on any model family.

F.4 Contamination control

BrowserBench sites and task templates are excluded from D_g, D_u and the online task pool. In addition, the online RL curriculum is composed from an early BrowserBench snapshot's diagnostic slices (i.e. which capability tags were weak) rather than from its task instances, so no benchmark task is trained on.

General Agentic Benchmark Contamination and Transfer Analysis. To verify that gains on general agentic suites (Tau2-Bench, BFCL-v4, Claw-Eval in Table 3) reflect genuine tool-use grounding

rather than data memorization: (1) *Strict Corpus Isolation*: Neither Live-Browser-Set (D_g, D_u, D_r) nor the online RL task pool shares task specifications, APIs, or evaluation environments with Tau2-Bench, BFCL, or Claw-Eval. (2) *Mechanism of Cross-Domain Transfer*: The $+6.6\%$ average gain across general benchmarks originates primarily from enhanced schema validation and recovery behavior. Browser-centric training penalizes malformed function calls (Section 6.1) and enforces structured error handling, which directly lifts API syntax compliance in BFCL-v4 and multi-turn state-tracking stability in Tau2-Bench without requiring task-specific fine-tuning.

G Implementation Details

RUIC-SFT. Backbones: Qwen3.5-4B/9B/27B. Two epochs, LoRA (Hu et al., 2022) with rank 16, $\alpha = 64$, all linear layers as target modules. Cosine schedule, peak LR 5×10^{-5}, 10% linear warmup. Per-device batch 6, gradient accumulation 2 (effective 12 per device); 8 GPUs with sequence parallelism 4, giving data-parallel degree 2 and global batch 24. The mixture schedule of Eq. (2) is realized by a dynamic sampler over the three subsets; the consequence of introducing D_r before λ_2 appears in row 5 of Table 4.

DAO-GRPO. LoRA rank 8, $\alpha = 32$, all linear layers. 150 iterations; each update consumes 6 valid groups of $K = 6$ trajectories after discarding zero-variance groups ($\approx$2.2 extra groups sampled per iteration), i.e., 5,400 rollouts consumed by updates out of $\approx$7,400 generated. Rollout cap 50 steps; this is a throughput–coverage trade-off, not a deployment horizon limit; it fully covers about 83% of the training pool, and longer tasks are truncated with the last state treated as non-terminal so accumulated shaping progress is retained. Evaluation always uses the full 100-step budget. Shaping uses the undiscounted reward-to-go of Section 6.1 with $\Phi(s_1) = 0$ (no initial progress). Rollout state includes the current page observation, recent screenshot context, and browser-tool feedback. Trajectories that terminate abnormally or become unreachable are masked by setting their advantages to zero. A single on-policy update is taken per rollout batch; asynchronous returns admit at most one policy-version lag, and π_{old} is the log-probability recorded by the inference engine during rollout rather than recomputed by the training engine, with the clipped ratio of Eq. (8) absorbing the residual. We use clip $\varepsilon = 0.15$ and KL coefficient $\beta_{\text{KL}} = 0.02$ to the frozen RUIC-SFT reference. The shrinkage coefficient $\hat{\kappa}$ is estimated from the previous iteration's batch so that $w_{i,t}$ remains independent of the current action.

Behavioural corrections $f_{i,t}$. Beyond the key-point potential, four corrections enter $G_{i,t}$ via its causal sum (Section 6.1) at the step that triggers them: a repeated-action penalty (0.02 per consecutive repeat, at most three consecutive and at most 0.3 per trajectory); a recovery refund returning at most the penalty actually incurred (0.03 each, at most 0.09 per trajectory); a format penalty (0.03) when a call fails schema validation; and a termination adjustment ($+0.02$ for a correctly signalled completion, $-0.05\,(1 - R_i^{\text{term}})$ for premature termination). Unlike the potential these do not telescope, so they enter $G_{i,t}$ and therefore $\hat{A}_{i,t}^{\text{DAO}}$; across logged training batches they move the mean return by 0.3%–4.5% relative to R^{term}. They keep returns non-degenerate when rollouts share a binary outcome, as in Figure 8(c).

Asynchronous rollout scheduling. Online training runs against a sandbox pool with thousands of concurrently active browsers. At each iteration a batch of tasks is dispatched, each task is rolled out by an independent group of sandboxes in parallel, and completed groups stream back to the optimizer while new rollouts continue in the background. Sandboxes are recycled per episode for state isolation.

Judge. A single judge model (Qwen3.7-Plus, fixed version) serves all reward computation. It receives the recorded log of a completed trajectory—full action history and recent screenshots—and in one offline pass returns (i) the terminal score $R^{\text{term}} \in [0, 1]$ against the task's structured success criterion and (ii) per-step key-point marks $k(s_{i,t})$, from which $\Phi(s_{i,t})$ is computed directly. It is never queried during interaction and never called per step; reward computation costs one judge pass per trajectory, in contrast to process reward models that score every step or VinePPO-style methods that re-rollout from every intermediate state. Judge prompts and the scoring rubric are released with the code.

Evaluation harness. Built on a customized version of `browser-use`[1]. All models, ours and baselines, use the identical tool space, DOM simplification, and step budget.

[1] `https://github.com/browser-use/browser-use`

G.1 Interference measurement protocol

The directed alignment $\mathcal{I}_{A\to B}(\lambda)$ of Eq. (1) is estimated at 20 equally spaced checkpoints during RUIC-SFT. At each checkpoint θ_λ, we compute $\nabla\mathcal{L}_A(\theta_\lambda)$ on a probe batch of 128 examples drawn from subset A, take a single gradient step of size $\nu = 10^{-4}$, and evaluate $\mathcal{L}_B^{\text{val}}$ on a held-out validation set of 256 examples from subset B. Bootstrap confidence intervals are computed over 50 resampled probe batches; the probe step size is chosen so that measured alignments change by less than 10% when ν is halved or doubled.

Held-out split construction and the source of asymmetry. The held-out split of D_r consists of *natural* errors extracted from bootstrapping rollouts, whereas the training split is dominated by annotator-injected errors (Appendix D), so the two splits differ systematically in their off-path contexts. Direct gradient computation on saved checkpoints confirms $\langle\nabla\mathcal{L}_{D_r}^{\text{val}}, \nabla\mathcal{L}_{D_g}\rangle > 0$ while $\langle\nabla\mathcal{L}_{D_g}^{\text{val}}, \nabla\mathcal{L}_{D_r}\rangle < 0$ for checkpoints before λ_2, with the second inner product approaching zero thereafter—supporting the distributional reading of the sign asymmetry in Figure 5.

G.2 Schedule parameter sweep

Table 4 in the main text reports the six curriculum variants that inform the data-mixture design. The terminal mixture ratios $(p_g^*, p_u^*, p_r^*) = (0.5, 0.3, 0.2)$ were selected from a grid of 9 candidates spanning $p_r^* \in \{0.1, 0.2, 0.3\}$ and $p_u^* \in \{0.2, 0.3, 0.4\}$ with $p_g^* = 1 - p_r^* - p_u^*$; the reported configuration maximizes validation performance on our held-out development tasks. Phase boundaries follow the probe: on 9B we round the measured $(\lambda_1, \lambda_2) = (0.31, 0.68)$ to $(0.3, 0.7)$; on 4B and 27B we use the measured λ_2 (0.79 and 0.55) with $\lambda_1 = 0.3$.

H Proofs of Theoretical Results

We formalize the mathematical properties of DAO-GRPO below.

Assumption 1 (Independence and Causality)**.** *(i) Sibling rollouts within group g execute in isolated sandboxes conditionally independent given task g and external site state ξ; (ii) The progress potential $\Phi(s_{i,t})$ is strictly causal, depending only on the trace history up to step t; (iii) Canonical equivalence $h_{j,t} \simeq h_{i,t}$ depends only on prefix actions $\{a_u\}_{u<t}$.*

Proposition 1 (Unbiased Step-Level Gradient)**.** *Under Assumption 1, for any finite horizon T_{cap}, the leave-one-out baseline $B_{i,t}$ is conditionally independent of the decision $a_{i,t}$ given context $c_{i,t}$, since $S_{i,t}$ and $h_{i,t}$ depend strictly on historical actions $\{a_{i,u}\}_{u<t}$, and sibling returns $\{G_{j,\cdot}\}_{j\neq i}$ are mutually independent under sandbox isolation. Consequently, the centering term introduces zero expected gradient bias:*

$$\mathbb{E}_{a_{i,t}\sim\pi_\theta}\left[\nabla_\theta \log \pi_\theta(a_{i,t} \mid c_{i,t}) B_{i,t} \,\middle|\, c_{i,t}\right] = \mathbf{0}, \tag{9}$$

guaranteeing that the unnormalized step-level surrogate gradient matches the true policy gradient: $\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{T_i} \nabla_\theta \log \pi_\theta(a_{i,t} \mid c_{i,t})(G_{i,t} - B_{i,t})\right] = \nabla_\theta J_{T_{cap}}(\theta)$. The subsequent batch-level standardization by $\hat{\sigma}_g$ serves as an empirical variance regularizer matching standard GRPO practice.

Proof of Proposition 1. Fix trajectory i and step t, and condition on the logged context $c_{i,t}$. The baseline $B_{i,t} = w_{i,t}\bar{G}_{i,t}^{\text{pre}} + (1 - w_{i,t})\bar{G}_i^{\text{grp}}$ depends only on: (i) the prefix $h_{i,t}$, determined by $\{a_{i,u}\}_{u<t}$ and hence fixed given $c_{i,t}$; (ii) sibling returns $G_{j,\cdot}$, $j \neq i$, which are independent of $a_{i,t}$ by sandbox isolation (Assumption 1(i)); and (iii) $\hat{\kappa}$, estimated from the previous iteration's batch (Appendix G). Thus $B_{i,t} \perp a_{i,t} \mid c_{i,t}$, and by the score-function identity,

$$\mathbb{E}_{a_{i,t}\sim\pi_\theta}\big[\nabla_\theta \log \pi_\theta(a_{i,t} \mid c_{i,t})\, B_{i,t} \,\big|\, c_{i,t}\big] = B_{i,t}\, \nabla_\theta \textstyle\sum_a \pi_\theta(a \mid c_{i,t}) = 0.$$

Moreover $G_{i,t}$ is a causal target: $R_i^{\text{term}} + \sum_{u\geq t} f_{i,u}$ is the reward-to-go from step t, and $\Phi(s_{i,t})$ is constant given $c_{i,t}$ since $s_{i,t}$ is determined by $\{a_{i,u}\}_{u<t}$ (Assumption 1(ii)). The claim then follows from the policy gradient theorem applied to the reward-to-go decomposition over the finite horizon T_{cap}. □

Proposition 2 (Variance Reduction at Divergence Points). *Write $\mu_h = \mathbb{E}[G_t \mid h]$, $\mu = \mathbb{E}[G_t]$, and total variance $\sigma^2 = Var(G_t)$. By the law of total variance, $\sigma^2 = \tau^2 + \sigma_w^2$, where $\tau^2 = Var_h(\mathbb{E}[G_t \mid h])$ denotes the between-cohort variance and $\sigma_w^2 = \mathbb{E}_h[Var(G_t \mid h)]$ denotes the expected within-cohort variance (matching the empirical batch estimators $\hat{\tau}^2$ and $\hat{\sigma}_w^2$ introduced in Section 6.2). At a divergence point with cohort size $n_{i,t} \geq 2$, the prefix LOO baseline attains lower MSE than the group LOO baseline whenever $(\mu_h - \mu)^2 > \sigma_w^2/(n_{i,t} - 1) - \sigma^2/(K - 1)$, i.e., when the between-branch outcome gap exceeds the within-cohort sampling noise. Moreover, the shrinkage weight $w_{i,t}$ in Eq. (6) represents the classical Bühlmann credibility solution, which is asymptotically MSE-near-optimal under empirical estimates $\hat{\kappa} = \hat{\sigma}^2_{within}/\hat{\tau}^2_{between}$ tracked from preceding batches.*

Proof of Proposition 2. Fix a prefix h with cohort size $n = n_{i,t} \geq 2$. As estimates of μ_h, the prefix LOO baseline is unbiased with $\mathrm{MSE}(\bar{G}^{\mathrm{pre}}) = \sigma_w^2/(n-1)$, while the group LOO baseline carries $\mathrm{MSE}(\bar{G}^{\mathrm{grp}}) = (\mu_h - \mu)^2 + \sigma^2/(K-1)$, the squared bias plus sampling variance; comparing the two gives the stated condition. For the blended estimator $\hat{\mu}_h(w) = w\,\bar{G}^{\mathrm{pre}} + (1-w)\,\bar{G}^{\mathrm{grp}}$, minimizing $\mathbb{E}[(\hat{\mu}_h(w) - \mu_h)^2]$ over w is the classical Bühlmann credibility problem, whose optimum from $n-1$ sibling observations is $w^* = (n-1)/(n-1+\kappa)$ with $\kappa = \sigma_w^2/\tau^2$; Eq. (6) plugs in the empirical estimates $\hat{\sigma}^2_{\mathrm{within}}, \hat{\tau}^2_{\mathrm{between}}$. □

Proposition 3 (Consistent Graceful Degradation). *As $n_{i,t} \to 1$ (complete rollout divergence) or when $\hat{\kappa} \to \infty$, $w_{i,t} \to 0$ and $B_{i,t} \to \bar{G}_i^{\mathrm{grp}}$. DAO-GRPO then reduces to group-relative optimization with a leave-one-out group baseline and a state-dependent potential control variate, inheriting the convergence properties of GRPO.*

Proof of Proposition 3. If $n_{i,t} = 1$, then $\max(0, n_{i,t} - 1) = 0$, so $w_{i,t} = 0$ and $B_{i,t} = \bar{G}_i^{\mathrm{grp}}$; the same limit holds as $\hat{\kappa} \to \infty$. The advantage reduces to the reward-to-go $G_{i,t}$ centered by a leave-one-out group baseline, with $-\Phi(s_{i,t})$ acting as a state-dependent control variate. Both are independent of $a_{i,t}$ given $c_{i,t}$ (Proposition 1), so the estimator coincides with group-relative policy optimization up to an action-independent additive constant per step. □

I Baseline References

Closed-source: GPT-5.5 (OpenAI, 2026), Ernie-5.0 (Wang et al., 2026a), Seed2.1-Pro (ByteDance Seed, 2026). Judge model: Qwen3.7-Plus. Open-source browser/general: OpenWebRL-8B (Yang et al., 2026), Kimi-k3 (Team et al., 2026a), Hy4-Preview (Tencent, 2026), Qwen3.5-4B/9B/27B (Yang et al., 2025), Qwen3.8-Max-xhigh-0902, DeepSeek-V4-Flash (Xu et al., 2026a). Agentic transfer: OpenCUA-72B (Wang et al., 2026b), GUI-Owl-1.5-32B-Instruct (Xu et al., 2026b), UI-Venus-1.5-30B-A3B (Team et al., 2026b), EvoCUA-32B-20260105 (Huang et al., 2026), Qwen3.5-27B (Yang et al., 2025), Qwen-UI-Agent-27B (Zhou et al., 2026).

J Reliability of the Progress Estimator Φ

We validate Φ against human judgment on 150 trajectory prefixes from held-out dev rollouts, each rated by two annotators on a five-level progress rubric anchored by subgoal descriptions (inter-annotator agreement 93% within one level, Spearman $\rho = 0.89$). Φ tracks human ratings closely (Spearman $\rho = 0.81$, mean absolute error 0.11 on the unit interval), agrees with the human progress ordering on 86% of 200 same-task prefix pairs, and is monotone non-decreasing along 87% of adjacent within-trajectory steps. The sign of the reward-to-go $G_{i,t}$ (Section 6.1) depends on this ranking property, and the potential enters the step-level advantage $\hat{A}_{i,t}^{\mathrm{DAO}}$ (Eq. 7) through both the individual return and the leave-one-out baseline, so calibration quality affects credit-assignment precision. Extending this validation to weaker judge models is a natural next step.

K Additional Results

K.1 Rollout-group case studies

These three groups illustrate why each DAO-GRPO component matches the structure of browser-agent data. (a) shows the shared-prefix/single-branch structure that motivates prefix-attributed credit:

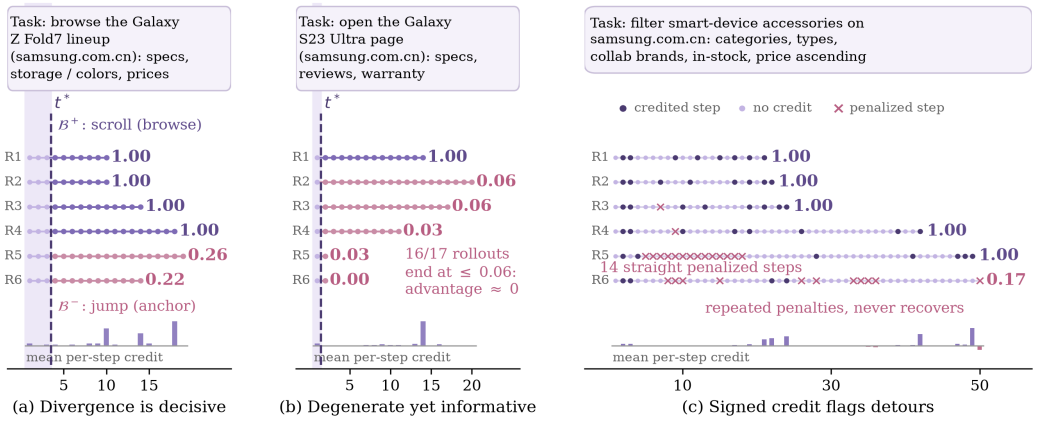


Figure 8: Real rollout groups from online RL training illustrating the three design choices of DAO-GRPO. For analysis we sample 16–20 rollouts per task (beyond the training group size $K{=}6$) and display six representative ones; row ends show final rewards and bottom strips show mean per-step reward-to-go $G_{i,t}$. **(a)** A Galaxy Z Fold7 browsing group: all rollouts share an identical three-step prefix (shaded corridor) and diverge at step 4 into top-down browsing (via `scroll`) vs. anchor jumping (via `send_keys` to the product anchor); 12 of 16 browsing rollouts succeed while 3 of 4 jumping rollouts stall at partial credit. **(b)** A Galaxy S23 Ultra group in which 16 of 17 rollouts fail: under a binary terminal reward the sixteen failures are indistinguishable from one another (all mapped to 0) and only the single success carries signal, whereas the continuous terminal score and judge-marked key-point increments order them ($1.00 > 0.06 > 0.03 > 0.00$) and localise where each stalled. **(c)** An accessory-filtering group where 12 of 16 rollouts all reach reward 1.0 with lengths from 21 to 49 steps; the behavioural corrections $f_{i,t}$ (repeated-action penalties, crosses) keep the group non-degenerate, e.g. 14 consecutive penalised steps in R5, separating efficient from wasteful executions that identical terminal rewards cannot distinguish.

rollouts are identical for three steps and the branch choice at one step largely determines the outcome, so shaped credit should concentrate on post-branch steps of the favorable branch. (b) shows a group whose sixteen failures a binary terminal reward cannot distinguish, but whose key-point increments order them. (c) shows that identical terminal rewards can hide a $2\times$ efficiency gap; the behavioural corrections keep such groups non-degenerate and order rollouts by efficiency.

K.2 Qualitative trajectory: detection and recovery

The task requires entering the Chery website, reaching the privacy-policy page, locating the update announcement, opening the personal-information collection list, and inspecting collected categories and purposes. The trajectory exhibits a perceive–act–verify–recover loop rather than monotonic execution. Steps 1–5 load the site, dismiss the cookie banner, scroll to the footer, and enter the privacy page. At Step 7 the agent clicks a misleading footer link and is silently redirected to an unrelated brand-history page; rather than assuming progress from having issued the action, it verifies the resulting state against the goal and at Step 8 detects the mismatch from URL and page content, then recovers by re-entering the privacy path. When repeated scrolling still fails (Steps 9–10), it escalates from low-cost navigation to structured page probing before retrying the correct route, eventually reaching the detailed policy page and completing the inspection (Steps 12–15).

This is consistent with the complementary roles of the two stages. D_r and RUIC-SFT teach the agent to judge progress from observed outcomes rather than from action issuance. DAO-GRPO strengthens the behavior online: here the successful and unsuccessful paths share a long prefix and only a few branch decisions determine whether the target document is reached, which is exactly what prefix-attributed credit targets; and the response-level objective optimizes each decision under its actual reconstructed context, helping the model rely on current page state, URL and visual evidence when deciding whether to continue, recover, or escalate.

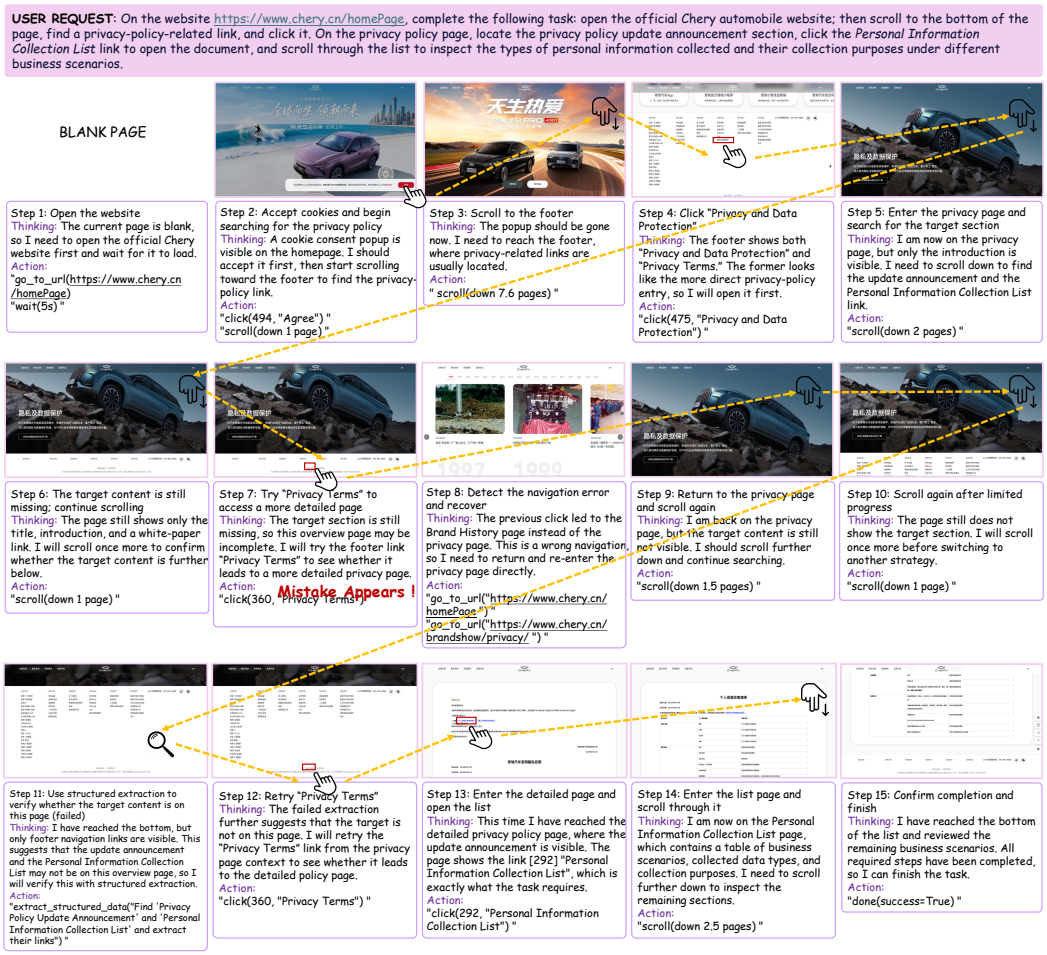


Figure 9: Self-reflective error recovery in long-horizon web navigation: a 15-step trajectory of Live-Browser-Agent-9B on a Chery privacy-policy retrieval task. The agent detects an erroneous redirect from the observed browser state, recovers, and escalates from scrolling to structured probing before reaching the personal-information collection list. Screenshots are shown for all steps for visualization; in deployment the agent primarily consumes structured browser state, with screenshots used only when visual grounding is needed.

L LIMITATIONS

Live-Web Nondeterminism. All live web evaluations are subject to website content drift and external service outages. While BrowserBench tasks are verified against frozen staging snapshots, evaluating on production websites over extended periods inevitably introduces stochastic variability. To mitigate this, we timestamp all benchmark runs and release our offline replay suite.

Sandbox Horizon Caps. For throughput efficiency in parallel online RL, training rollouts are capped at $T_{\text{cap}} = 50$ steps. Figure 6 suggests transfer to longer horizons at test time, though the $>$50-step bucket contains few tasks; workflows requiring more than 50 steps receive terminal partial credit rather than full exploratory rollouts during training. Relaxing this horizon cap via dynamic batching is a natural direction for future engineering scaling.

Prefix Cohort Thinning. In open-ended exploratory tasks with extremely divergent paths, the size of prefix cohort $|S_{i,t}|$ naturally decays as execution depth increases. Although our empirical Bayes shrinkage guarantees smooth degradation to standard group baselines ($w_{i,t} \to 0$), the advantage localization signal is most concentrated in the initial and middle phases of trajectories where branches cleanly diverge.

Linguistic and Regional Coverage. While BrowserBench features balanced Chinese and English tasks across 254 production websites, it does not currently cover right-to-left scripts or localized intranet workflows. Expanding task diversity across broader regional domains remains important future work.