UNCURL: RELIABILITY-CONDITIONED PARTIAL SHARING FOR JOINT CURRICULUM AND EXPLORATION CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

Efficiently training large language model (LLM) agents requires deciding both which tasks to practice and how much to explore within each task. Existing controllers usually estimate these decisions independently, although both can depend on shared capabilities; forcing one shared uncertainty representation, however, can propagate proxy error and cause negative transfer. Our contributions are threefold. We formulate curriculum–exploration coordination as conditional partial sharing between task-level learning progress (LP) and state-action information gain (IG). We introduce **UnCurl**, which retains private LP and IG factors and admits a shared predictive factor only when calibrated dependence, reliability, held-out utility, effective sample size, and drift tests pass. A monotone soft gate controls sharing strength, while an independently trained shadow posterior provides fail-closed fallback. On ALFWorld, UnCurl reaches 91% unseen success versus 86% for interaction-matched INDEPENDENT LP+IG, raises normalized AULC from 73 to 81, and reduces interactions to the 80% development threshold from 43k to 30k. It lowers harmed-task fraction from 0.48 under FULL-SHARE LP+IG to 0.16 and reaches 44% success on WebArena-Lite. UnCurl uses 37.7% more GPU-hours than INDEPENDENT LP+IG.

1 INTRODUCTION

Training language agents requires two coupled but distinct decisions: **which tasks to practice** and **how to explore within each task**. This allocation problem recurs across web navigation, software engineering, and machine-learning workflows, where agents face heterogeneous tasks, long-horizon interaction, and expensive feedback (Zhou et al., 2024; Yang et al., 2024; Chan et al., 2025). Curriculum methods allocate training effort using difficulty, failures, or expected improvement (Rigter et al., 2024; Zhang et al., 2024; Qi et al., 2025; Lai et al., 2024; Xi et al., 2025); exploration methods allocate actions, temperatures, or resampling using uncertainty (Wang et al., 2026; Zhang et al., 2026; Wei et al., 2025; Bai et al., 2026). Independent controllers are appropriate when their signals are unrelated, but can duplicate statistical effort when task-level learning progress and state-action information gain reflect common capabilities.

Unconditional sharing is also unsafe. LP estimates expected post-update improvement, whereas IG estimates reducible state-action uncertainty. They differ in granularity, semantics, and decision objective. Their practical proxies are noisy, correlation can be nonlinear, and common trajectories can induce correlated errors. High uncertainty can moreover be irreducible. Consequently, neither observed agreement nor one favorable dependence test justifies sharing.

The ambiguity is statistical as well as architectural. Weak observed dependence can mean that the underlying signals are unrelated, but it can also result from attenuation by measurement error or insufficient effective sample size. Strong observed dependence is not sufficient either: two proxy heads can agree because they share data, backbone features, or a systematic failure mode. A useful controller must therefore answer three questions separately: whether dependence is detectable, whether sharing improves held-out prediction, and whether that conclusion remains stable as the policy changes the data distribution.

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

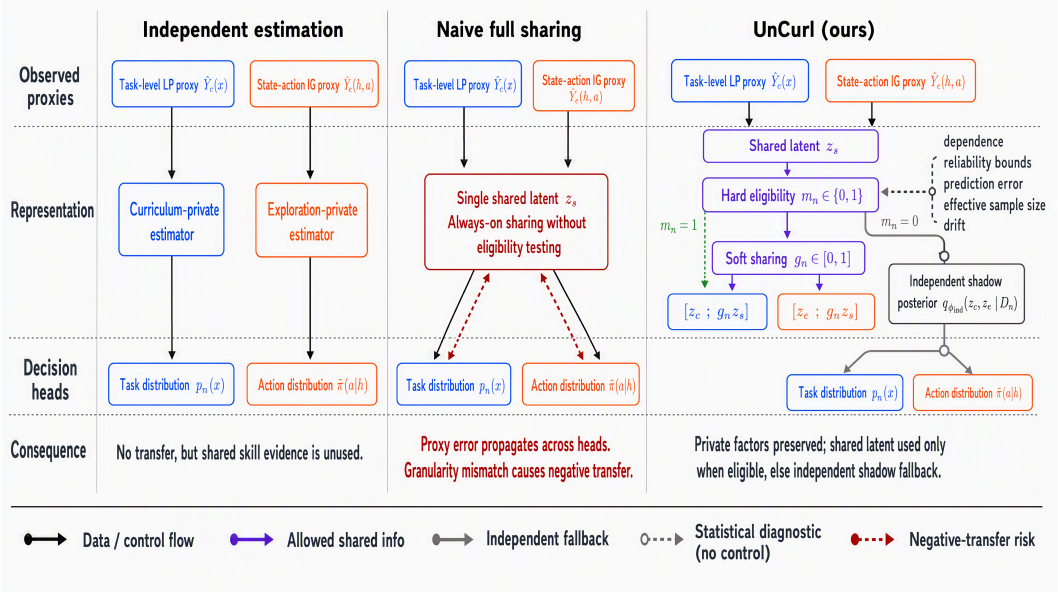


Figure 1: Representation-sharing alternatives. Independent controllers cannot exploit common predictive factors; naive full sharing propagates proxy error. UnCurl retains private factors, admits shared information through hard eligibility m_n and soft gate g_n, and routes ineligible rounds to an independently trained shadow posterior.

We propose **UnCurl**, a reliability-conditioned controller for a fixed task pool. It separates measurement, representation, and control. A joint posterior contains shared, curriculum-private, and exploration-private predictive factors, but a hard eligibility test and a soft gate determine whether shared information reaches the two decision heads. If evidence is weak, unreliable, underpowered, or shifted, both heads switch to an independently trained shadow posterior. Here "reliability-conditioned" means that estimated proxy reliability is an input to this routing decision. The controller therefore treats coupling as a falsifiable decision rather than an architectural assumption.

Figure 1 summarizes the design space: UnCurl retains private paths and activates a shared path only when calibrated diagnostics pass.

Our contributions are:

1. We formulate curriculum–exploration coordination as conditional partial sharing between distinct task-level LP and state-action IG estimands.
2. We introduce a calibrated hard mask, monotone soft gate, and independent fallback driven by dependence, reliability, held-out prediction, effective sample size, and drift.
3. Controlled dependence sweeps, interaction-matched ALFWorld training, and held-out WebArena-Lite evaluation show improved learning speed, generalization, and negative-transfer control while exposing the additional compute cost.

2 Problem and Method

Figure 2 gives the complete dependency chain. First, LP and IG define the two decision targets. Second, controlled problems calibrate their measurement reliability and the power of dependence tests. Third, a partially shared posterior produces private and common predictive factors. Fourth, cross-granularity evidence and drift determine sharing eligibility and strength. Finally, the routed factors drive task selection and action exploration, whose outcomes update the next controller round.

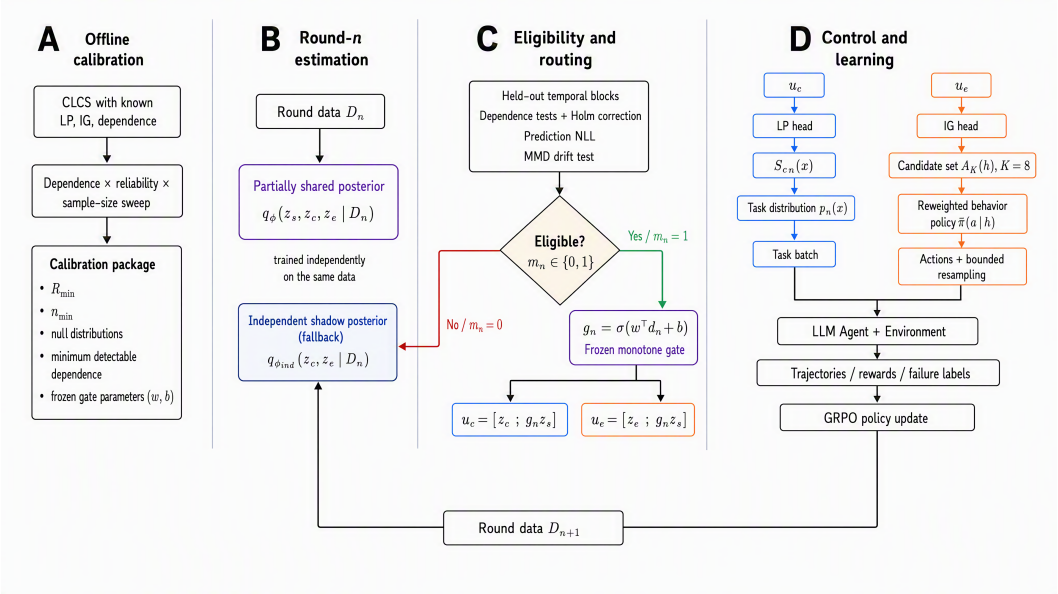


Figure 2: UnCurl training and routing: (1) estimate LP/IG targets, (2) calibrate reliability and detection power, (3) fit shared and independent posteriors, (4) compute eligibility m_n and sharing strength g_n, and (5) select task batches and within-task actions. Outcomes feed the next controller round.

2.1 Decision Targets and Objective

Let $\mathcal{X}$ be a finite task pool. Task x induces a finite-horizon POMDP, and policy $\pi_\theta(a_t \mid h_t, x)$ acts from interaction history h_t. At round n, a curriculum controller selects task distribution $p_n(x)$, while an exploration controller allocates a fixed within-trajectory budget. The ideal curriculum signal is expected post-update progress,

$$\mathrm{LP}_n(x) = \mathbb{E}[J_{\mathrm{dev}}(\theta'_x) - J_{\mathrm{dev}}(\theta) \mid \mathcal{D}_{\le n}],$$

and the ideal exploration signal is posterior information gain,

$$\mathrm{IG}_n(a \mid h) = H[q(Z_e \mid \mathcal{D}_{\le n})] - \mathbb{E}_o H[q(Z_e \mid \mathcal{D}_{\le n} \cup \{h, a, o\})].$$

LP is estimated by a gain predictor selected against bias-corrected EMA differences on held-out one-step gain probes. IG is approximated using five bootstrapped posterior heads and four simulated outcome updates. Between-head mean variance is epistemic; mean predictive variance is aleatoric. Full definitions and calibration checks are in Appendix B.

Problem definition. Given interaction budget B, UnCurl learns a curriculum p_n, an exploration allocation ξ_n, and policy parameters θ_n:

$$\max_{\theta, \{p_n, \xi_n\}} \sum_{n=1}^{N} J_{\mathrm{dev}}(\theta_n) \quad \text{s.t.} \quad \sum_{n=1}^{N} \mathrm{Cost}(p_n, \xi_n) \le B.$$

The development split supplies delayed LP targets. A separate validation split selects controller checkpoints and global hyperparameters; seen and unseen test tasks are never queried by either controller. The design imposes three requirements: task selection remains driven by LP and action exploration by IG; a null dependence estimate is actionable only inside a calibrated power and reliability region; and failed diagnostics must route to an independently trained estimator. These two noisy target streams are the inputs to the calibration and latent-estimation stage below.

2.2 Proxy Calibration and Latent Estimation

Calibration before control. We construct latent contextual bandits and small MDPs with

$$z_c = A_c z_s + \varepsilon_c, \qquad z_e = A_e z_s + \varepsilon_e,$$

and vary A_c, A_e, noise covariance, and sample size to create negative, null, weak, and strong dependence with known ground truth. For dependence δ, reliabilities R_c, R_e, and sample size n, calibration estimates

$$\mathcal{P}_{\text{det}}(\delta, R_c, R_e, n) = \Pr(\text{reject } H_0 \mid \text{Dep}(Y_c, Y_e) = \delta).$$

The minimum detectable dependence $\delta_{\min}$ is the smallest value with power at least 0.8 at the registered false-positive rate. Consequently, a null on agent traces means only that no useful dependence above $\delta_{\min}$ was detected. If reliability falls outside calibrated support, sharing is disabled rather than interpreting the null as independence.

Reliability estimation. For noisy proxy $\hat{Y}_j$ and probe target Y_j^{probe},

$$R_j = \left[1 - \frac{m_j^{-1} \sum_i (\hat{Y}_{j,i} - Y_{j,i}^{\text{probe}})^2}{\widehat{\text{Var}}(Y_j^{\text{probe}}) + \epsilon_R}\right]_0^1, \qquad j \in \{c, e\}.$$

Curriculum reliability uses 32 one-step probes per round. Exploration reliability transfers a bootstrap lower bound from a CLCS lookup indexed by ensemble calibration and repeated-state prediction errors; out-of-support observations are ineligible. Reliability therefore determines whether the observed LP/IG streams can enter the shared-structure test.

Partially shared posterior. Using the observed LP/IG streams, UnCurl fits

$$q_\phi(z_s, z_c, z_e \mid \mathcal{D}_{\le n}),$$

where z_s is shared and z_c, z_e are private predictive factors. Task-conditioned LP, state-conditioned IG, and outcome likelihoods are jointly optimized with equal per-observation weight. The independent baseline removes z_s; full sharing removes private factors, with widths reallocated to match parameters. The posterior converts calibrated observations into latent traces for the subsequent cross-controller test; Appendix C gives the full model.

2.3 Cross-Granularity Dependence and Drift

Common statistical unit. LP is task-round level, while IG is state-action level. For task x in temporal block b, we aggregate

$$\bar{Y}_{e,b}(x) = \frac{\sum_{(h,a)\in\mathcal{T}_b(x)} \omega_{h,a} \widehat{\text{IG}}(a \mid h)}{\sum_{(h,a)\in\mathcal{T}_b(x)} \omega_{h,a}},$$

using inverse-frequency weights. The statistical unit is therefore $(\widehat{\text{LP}}_b(x), \bar{Y}_{e,b}(x))$.

Dependence evidence. On this common unit, the pre-registered evidence vector combines partial correlation, conditional mutual information, HSIC, CCA, and bidirectional held-out cross-prediction. Each statistic is standardized with 1,000 block permutations. Holm correction controls the six tests within a block, and Pocock alpha spending controls repeated tests across blocks. A test enters $\mathcal{K}_n$ only if it remains significant and has at least 0.8 calibrated power.

Non-directional statistics establish the presence of dependence but do not determine whether sharing helps. We therefore additionally require both $\Delta R^2_{c\leftarrow e}$ and $\Delta R^2_{e\leftarrow c}$ to be positive on held-out blocks. Significant, powered tests are aggregated as

$$\widehat{\text{Dep}}_n = \begin{cases} \text{median}_{k\in\mathcal{K}_n} z_{k,n}, & |\mathcal{K}_n| \ge 3, \\ 0, & \text{otherwise.} \end{cases}$$

The threshold of three prevents one favorable statistic from activating sharing. Effective sample size is reduced for positive temporal autocorrelation and uses the minimum value over projected dimensions.

Distribution stability. Drift is measured with a frozen representation,

$$\delta_n = \text{MMD}^2_\kappa(\psi_{\text{anchor}}(\mathcal{B}_n), \psi_{\text{anchor}}(\mathcal{B}_{n-w:n-1})),$$

and a one-sided equivalence test of $H_0 : \delta_n \ge \delta_{\max}$ yields $p_{\text{eq},n}$ under a separate sequential alpha budget. Dependence, bidirectional predictive utility, reliability, effective sample size, and drift now form the complete input to the routing decision.

2.4 Adaptive Sharing Decision

Hard eligibility and soft sharing. The hard mask is

$$m_n = \mathbb{I}[|\mathcal{K}_n| \geq 3,\ \Delta R^2_{c\leftarrow e} > 0,\ \Delta R^2_{e\leftarrow c} > 0,\ \underline{R}_{c,n}, \underline{R}_{e,n} \geq 0.50,\ n_{\text{eff}} \geq n_{\min},\ p_{\text{eq},n} < \alpha_n^{\text{drift}}].$$

Conditional on eligibility,

$$g_n = m_n \sigma\Big(\mathbf{w}^\top [\widehat{\text{Dep}}_n, R_{c,n}, R_{e,n}, -e_{c,n}, -e_{e,n}, -\delta_n, \log n_{\text{eff}}] + b\Big).$$

The gate is calibrated only in CLCS. For each held-out calibration cell r, its target is the discrete sharing level

$$g_r^\star = \arg\min_{g\in\{0,.25,.5,.75,1\}} \tfrac{1}{2}\left[\mathcal{L}^{\text{val}}_{\text{LP},r}(g) + \mathcal{L}^{\text{val}}_{\text{IG},r}(g)\right],$$

where the two losses are standardized predictive negative log likelihoods. The monotone gate minimizes $\sum_r (g_\psi(v_r) - g_r^\star)^2$, where v_r contains dependence, reliability, error, drift, and effective sample size. Thus, the gate target operationalizes *reliability-conditioned predictive sharing*; it does not optimize ALFWorld or WebArena success. Nonnegative feature weights make sharing monotone in dependence, reliability, negative error, and negative drift. Gate parameters are frozen before agent training. The two representations are

$$u_{c,n} = [z_c; g_n z_s], \qquad u_{e,n} = [z_e; g_n z_s].$$

When $m_n = 0$, both controllers use the parameter-matched independent shadow posterior rather than private features co-trained through the shared path. Appendix C gives the full posterior, test battery, effective-sample-size definition, and gate calibration.

The gate also scales the residual posterior coupling, $\Sigma^{(n)}_{ce}(g_n) = g_n \Sigma^{(n)}_{ce}$. The shadow posterior receives the same accumulated observations and number of updates as the shared model. This detail is necessary for a valid fallback: setting $g_n = 0$ in a previously co-trained representation would not recover the independent estimator used by the primary baseline. The output of this stage is therefore either gated shared/private representations or fully independent shadow representations; only these routed features are exposed to the controllers. This separation prevents downstream leakage: agent outcomes evaluate whether proxy-level reliable sharing transfers to policy utility, rather than supplying the gate supervision.

2.5 Curriculum and Exploration Control

Given routed representations, the two heads predict $\widehat{\text{LP}}_n(x) = f_c(u_{c,n}, e_x)$ and $\widehat{\text{IG}}_n(a \mid h) = f_e(u_{e,n}, h, a)$. The first output determines which task is practiced; conditional on that task, the second determines how the fixed action budget is allocated.

Curriculum head. The curriculum head scores

$$S_{c,n}(x) = \widehat{\text{LP}}_n(x) - \lambda_a \widehat{U}_{\text{ale}}(x) + \lambda_d \,\text{Div}_n(x) + \lambda_u \,\text{UCB}_n(x), \quad p_n(x) \propto e^{S_{c,n}(x)/\tau_c}.$$

Exploration head. Given the selected task, the exploration head scores

$$S_{e,n}(a \mid h) = \widehat{\text{IG}}_n(a \mid h) - \mu_a \widehat{U}_{\text{ale}}(a \mid h) - \mu_c \,\text{Cost}(a).$$

It reweights a fixed greedy/nucleus/admissible candidate set using length-normalized policy likelihood and $S_{e,n}$, then allocates at most two additional rollouts to the top state decile. First-pass rollouts are reduced to keep policy actions and environment calls fixed. Appendix D gives the sampling distribution, duplicate handling, and behavior likelihood.

Coordinated updates. Training begins with round-robin coverage. At each round, UnCurl updates gain probes, fits the shared and shadow posteriors, computes dependence and drift, sets m_n, g_n, samples task batches, allocates bounded exploration, updates the policy with environment outcome reward, and records delayed progress for the next round. Task generation is deliberately excluded so that improvements can be attributed to task selection, exploration, and representation sharing. The delayed LP and outcome records then return to the calibration/posterior stage, closing the loop in Figure 2.

2.6 Risk Analysis

The sharing gate is motivated by the following bias–variance trade-off. For scalar estimators $\hat{\theta}_c \sim \mathcal{N}(\theta_c, v_c)$ and $\hat{\theta}_e \sim \mathcal{N}(\theta_e, v_e)$, noise covariance κ_{ce}, and $\hat{\theta}_c(\alpha) = (1-\alpha)\hat{\theta}_c + \alpha\hat{\theta}_e$,

$$\mathcal{R}_c(\alpha) - \mathcal{R}_c(0) = -2\alpha(v_c - \kappa_{ce}) + \alpha^2(v_c + v_e - 2\kappa_{ce} + \Delta_{ce}^2),$$

where $\Delta_{ce} = \theta_e - \theta_c$. If $v_c > \kappa_{ce}$, sharing improves risk exactly for

$$0 < \alpha < \frac{2(v_c - \kappa_{ce})}{v_c + v_e - 2\kappa_{ce} + \Delta_{ce}^2}.$$

Thus useful sharing requires variance reduction to exceed correlated error and transfer bias. The complete proposition, proof, and scope are in Appendix E. The result does not assert a universal nonlinear threshold; it identifies why dependence alone is insufficient and why proxy error, transfer bias, and correlated noise must enter the operational gate.

3 Experiments

Evaluation questions. The study asks whether the proxies identify shared structure (RQ1), whether adaptive sharing improves final performance and learning efficiency over interaction-matched Independent LP+IG and Full-share LP+IG controls (RQ2), whether fallback protects the lower tail under weak dependence and drift (RQ3), and whether the effect transfers to WebArena-Lite (RQ4). Agent results aggregate five independent runs; each CLCS grid cell contains 200 independently generated problems.

3.1 Experimental Setup

Setup. The Calibrated Latent Control Suite (CLCS) varies true dependence from -0.8 to 0.8, proxy reliability from 0.2 to 1.0, and effective sample size from 64 to 2,048, with 200 problems per cell. ALFWorld uses 3,553 training, 140 seen-test, and 134 unseen-test games across six families. Training IDs are split 70/10/10/10 into policy training, controller development, gain probes, and hyperparameter validation. Official tests are never queried by a controller. External evaluation uses the 165-task VAB-WebArena-Lite suite with fresh site snapshots (Zhou et al., 2024; Koh et al., 2024; Xie et al., 2024). The remaining 647 WebArena tasks are not treated as an official training split; web adaptation uses released WebRL trajectories and generated instructions without access to the 165 evaluation tasks.

All controlled methods use the same BF16 Qwen3.5-9B initialization, LoRA targets, action grammar, SFT/RFT initialization, outcome reward, and GRPO recipe: learning rate 10^{-6}, prompt batch 64, eight rollouts, 150 updates, and a 50-action horizon. Five independent seeds are $\{17, 42, 2026, 31415, 65537\}$. Policy rollouts, interactions, optimizer updates, and policy-generated tokens are matched; total H100-hours additionally include probes, ensembles, posterior fitting, and failed rollouts.

The posterior uses a 512-dimensional task/state encoder, 128-dimensional shared latent, and two 64-dimensional private latents. The shadow posterior has the same total latent width and update count. Within each seed, methods share the initial policy checkpoint, candidate-pool random stream, and evaluation episodes, but independently sample training trajectories and minibatches. This paired design reduces nuisance variation without coupling policy updates.

The primary comparator is Independent LP+IG, which shares all estimators, features, policy settings, and interaction budgets without representation sharing. Other controlled baselines are Uniform, LP-only, IG-only, Full-share LP+IG, Fixed-partial-share, WAKER-style acquisition, adapted WebRL, T^{2}PO, and SELAUR (Rigter et al., 2024; Qi et al., 2025; Wang et al., 2026; Zhang et al., 2026). Native-setting systems remain contextual because their data and systems budgets differ (Lai et al., 2024; Wei et al., 2025; Bai et al., 2026). Appendix F specifies every controlled adaptation.

Uniform uses uniform tasks and base exploration. LP-only changes only task selection; IG-only changes only action exploration. Full-share LP+IG removes private latents and keeps sharing

Table 1: Controlled Qwen3.5-9B comparison over five runs. Entries are rounded seed means; primary paired contrasts and 95% CIs are reported in text. Steps is thousands of interactions to 80% development SR; H100 h includes controller cost.

Method	Seen	Unseen	Macro	AULC	Steps	Web SR	H100 h
Uniform + fixed exploration	79.2	76.4	77.6	59.3	73,3	29.5	1,100
LP-only curriculum	82.0	81.4	81.2	65.6	62.5	35.5	1,225
IG-only exploration	84.2	83.5	83.6	68.0	55.2	34.1	1,350
Independent LP+IG	88.1	86.1	87.4	73.7	43.5	39.8	1,525
Full-share LP+IG	86.5	83.3	83.1	69.3	53.0	37.2	1,475
WAKER-style	83.3	82.7	82.2	66.3	58.6	33.3	1,350
WebRL adapted	86.1	84.4	85.2	70.6	48.4	43.3	1,625
T^2PO controlled	91.1	90.2	90.0	77.4	35.6	40.3	1,450
SELAUR controlled	90.2	88.4	89.5	75.5	38.0	39.2	1,400
Fixed-partial-share	90.1	88.6	89.3	77.0	36.2	41.3	1,850
Adaptive-share UnCurl	**92.3**	**91.5**	**91.3**	**81.6**	**30.1**	**44.2**	2,100

active. FIXED-PARTIAL-SHARE retains private latents but fixes $g = 0.5$ without a shadow switch. ADAPTIVE-SHARE UNCURL uses the complete diagnostics and fallback. WAKER-style, WebRL, T^2PO, and SELAUR retain their defining acquisition, curriculum, exploration, or reward mechanism under the common backbone and policy budget.

Metrics and inference. We report seen and unseen success rate (SR), unweighted six-family macro SR, normalized AULC, interactions to 80% controller-development SR, WebArena-Lite macro SR, and H100-hours. Unseen SR is mean terminal success over the 134 unseen games; Macro is the unweighted mean of the six family-level SRs. AULC integrates family-macro SR over 0–60k interactions and divides by 60,000. Steps is linearly interpolated from development evaluations and is reported only if at least four seeds cross. The two co-primary endpoints are unseen SR and AULC for ADAPTIVE-SHARE UNCURL relative to INDEPENDENT LP+IG; the corresponding contrasts with FULL-SHARE LP+IG are confirmatory.

For every agent contrast, the paired hierarchical bootstrap resamples the five training seeds, task families within each sampled seed, and task IDs within each sampled family for 10,000 replicates, preserving method pairing and recomputing macro SR, AULC, and their difference. We report unadjusted two-sided percentile 95% confidence intervals (CIs). Two-sided randomization p-values permute paired task outcomes within seed and family; Holm correction covers the four endpoint–comparator tests. Steps is treated as a censored descriptive measure, not a primary significance test. CLCS power, false-positive rate (FPR), and false-negative rate (FNR) use binomial Wilson 95% CIs over 200 replicates. We do not infer significance from overlap between separately computed intervals. WebArena-Lite and ablation comparisons are descriptive unless a paired contrast is explicitly specified.

3.2 MAIN RESULTS

Performance and learning efficiency. ADAPTIVE-SHARE UNCURL improves unseen SR from 86% to 91% over INDEPENDENT LP+IG, a paired difference of $+5.0$ pp $[+2.1, +7.9]$, and AULC from 73 to 81, a difference of $+8.0$ $[+3.6, +12.4]$. The corresponding two-sided tests remain significant after Holm correction at $\alpha = .05$; the displayed CIs are unadjusted. Against FULL-SHARE LP+IG, the corresponding differences are $+8.0$ pp $[+4.6, +11.4]$ and $+12.0$ $[+7.5, +16.5]$. On WebArena-Lite, ADAPTIVE-SHARE UNCURL reaches 44% versus 39% for INDEPENDENT LP+IG and 43% for adapted WebRL. This supports directional transfer under the tested web protocol, not a general ranking against native systems.

Table 2: Proxy calibration in medium-noise CLCS. Reliability is normalized predictive accuracy; $\delta_{\min}$ is dependence detected with 80% power.

Proxy	R^2	Kendall τ	Reliability	$\delta_{\min}$	FPR	FNR
EMA-LP	.24	.30	.48	.40	.06	.31
Gain predictor	.44	.46	.66	.28	.06	.21
Ensemble disagreement	.35	.41	.61	.31	.06	.24
Approximate IG	**.53**	**.55**	**.75**	**.23**	**.05**	**.16**

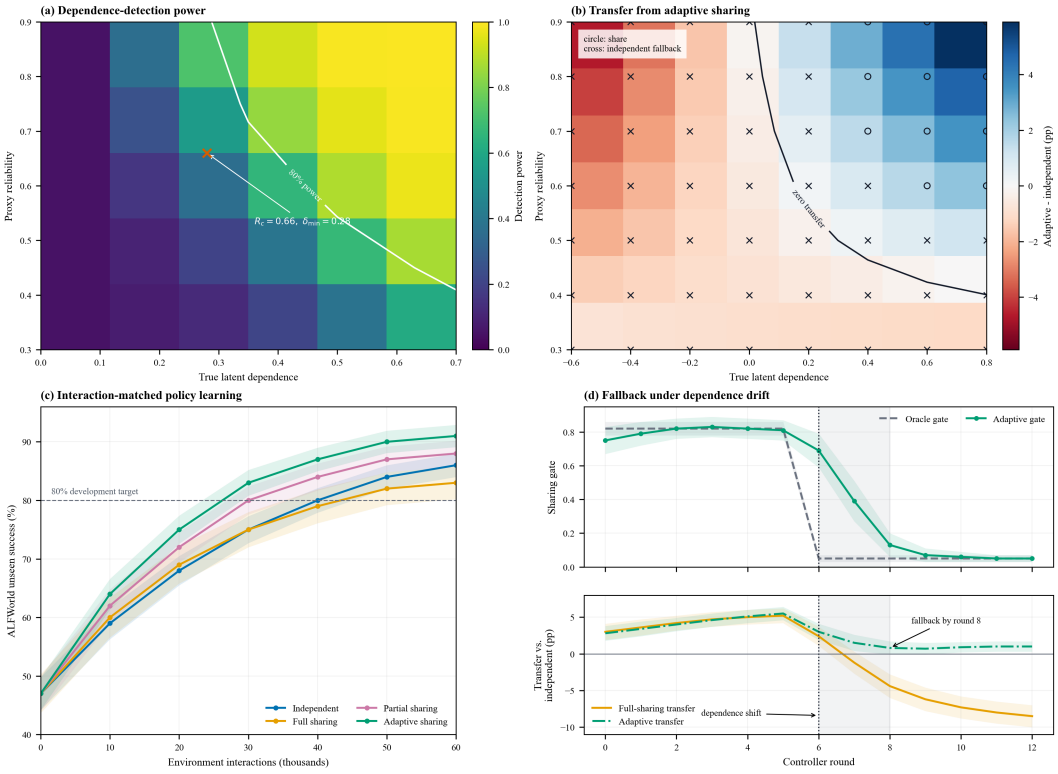


Figure 3: Calibration and mechanism results. (a) Dependence-detection power. (b) Adaptive-sharing transfer; circles share and crosses use fallback. (c) Interaction-matched ALFWorld unseen SR. (d) Gate and transfer after a dependence shift at round 6. Vertical bars in (c)–(d) show pointwise hierarchical-bootstrap 95% CIs.

3.3 Proxy Validity and Sharing Dynamics

Proxy validity and mechanism. In medium-noise CLCS, the gain predictor improves reliability from .48 to .66 and lowers minimum detectable dependence from .40 to .28 relative to EMA-LP. Approximate IG improves reliability from .61 to .75 and lowers FNR from .24 to .16 relative to raw disagreement. Figure 3(a) shows that power increases jointly with dependence and reliability. Panel (b) localizes positive transfer to the identifiable region.

After the round-6 dependence shift in panel (d), the adaptive gate approaches the oracle fallback by round 8. Additional gate diagnostics, including sharing eligibility, gate trajectories, and fallback statistics, are reported in Appendix H. Adaptive transfer stabilizes near +1.0 point, whereas forced FULL-SHARE LP+IG declines to -8.5 points. ADAPTIVE-SHARE UNCURL reduces harmed-task fraction from .48 under FULL-SHARE LP+IG to .16 and changes worst-family transfer from -8.5 to

+1.0 points. Relative to FIXED-PARTIAL-SHARE, it raises unseen SR from 88 to 91 and AULC from 77 to 81. Complete proxy, negative-transfer, and component tables are in Appendix G.

3.4 ABLATIONS AND EFFICIENCY

Removing calibration lowers unseen SR from 91 to 86 and AULC from 81 to 72, while FNR increases from .18 to .37. Removing reliability produces similar performance and increases harmed-task fraction from .16 to .39. A Pearson-only gate reaches 87 unseen SR and 74 AULC, showing that one linear statistic does not replace the calibrated evidence battery. Removing drift input reaches 88/74, indicating that static dependence evidence is insufficient after the policy changes its visitation distribution.

The uncertainty decomposition is also consequential. Replacing approximate IG with token entropy lowers unseen SR from 91 to 86 and AULC from 81 to 72. Removing the aleatoric penalty yields unseen SR of 86 and AULC of 71, while harmed-task fraction rises to .38. Thus, the observed gains require both conditional sharing and a distinction between reducible and irreducible uncertainty.

Compute. ADAPTIVE-SHARE UNCURL uses 2,100 H100-hours per run versus 1,525 for INDEPENDENT LP+IG, a 37.7% premium; interaction counts and H100 cost are reported separately.

4 RELATED WORK

Agent training and evaluation. Interactive agent research has progressed from prompting and supervised agent tuning to environment-grounded reinforcement learning. Agent-FLAN studies data and optimization choices for general agent tuning, while AgentGym provides a multi-environment training and evaluation framework (Chen et al., 2024; Xi et al., 2025). WebRL, WebAgent-R1, and AsyncWebRL develop online curricula, end-to-end multi-turn RL, and asynchronous systems for web agents (Qi et al., 2025; Wei et al., 2025; Bai et al., 2026). WebArena, VisualWebArena, OSWorld, SWE-Agent, and MLE-Bench broaden evaluation from browser navigation to computer use, software repair, and machine-learning engineering (Zhou et al., 2024; Koh et al., 2024; Xie et al., 2024; Yang et al., 2024; Chan et al., 2025). UnCurl complements these works by controlling how training tasks and within-task exploration share statistical evidence.

Curriculum and exploration. Automated curriculum learning, Prioritized Level Replay, WAKER, WebRL, and OMNI adapt task allocation to the learner (Graves et al., 2017; Jiang et al., 2021; Rigter et al., 2024; Qi et al., 2025; Zhang et al., 2024). RND and Plan2Explore motivate novelty and model disagreement (Burda et al., 2018; Sekar et al., 2020), while T^2PO and SELAUR adapt uncertainty control to agentic RL (Wang et al., 2026; Zhang et al., 2026). UnCurl differs by retaining LP and IG as distinct estimands and testing whether their representations should share.

Multi-task sharing. Multi-task learning can reduce variance but suffer negative transfer. Task-grouping and gradient-based methods study transfer affinity and interference (Standley et al., 2020; Wang et al., 2020). UnCurl addresses a non-stationary setting with different temporal granularities and makes dependence, reliability, utility, drift, and fallback separately testable. Extended positioning appears in Appendix I.

5 CONCLUSION

Adaptive UnCurl reaches 91% unseen ALFWorld success, raises AULC from 73 to 81, reduces Steps from 43k to 30k, lowers harmed-task fraction to .16, and reaches 44% WebArena-Lite SR. Across the evaluated protocols, conditioning sharing on dependence, proxy reliability estimates, held-out utility, sample size, and drift yields stronger results than the registered independent and fixed-sharing controls.

Reproducibility statement

The main text specifies the estimands, gate, data isolation, primary budgets, seeds, metrics, and tests. Appendix F records the complete protocol, and Appendix I lists the manifests, configurations, run ledger, and per-task records required to regenerate all tables and figures.

AI use statement

Generative AI tools were used to assist research ideation, language editing, and conversion of the authors' research notes into LaTeX. All AI-assisted text, mathematical claims, citations, and artifacts were reviewed by the authors, who take responsibility for the final content.

References

Hao Bai, Rui Yang, Chenlu Ye, Spencer Whitehead, Aviral Kumar, and Tong Zhang. Asyncwebrl: Efficient multi-step rl for visual web agents. *arXiv preprint arXiv:2606.05597*, 2026.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.

Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, et al. Mle-bench: Evaluating machine learning agents on machine learning engineering. In *International Conference on Learning Representations*, volume 2025, pp. 50466–50494, 2025.

Zehui Chen, Kuikun Liu, Qiuchen Wang, Wenwei Zhang, Jiangning Liu, Dahua Lin, Kai Chen, and Feng Zhao. Agent-flan: Designing data and methods of effective agent tuning for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 9354–9366, 2024.

Alex Graves, Marc G Bellemare, Jacob Menick, Remi Munos, and Koray Kavukcuoglu. Automated curriculum learning for neural networks. In *international conference on machine learning*, pp. 1311–1320. Pmlr, 2017.

Minqi Jiang, Edward Grefenstette, and Tim Rocktäschel. Prioritized level replay. In *International Conference on Machine Learning*, pp. 4940–4950. PMLR, 2021.

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.

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.

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.

Marc Rigter, Minqi Jiang, and Ingmar Posner. Reward-free curricula for training robust world models. In *International Conference on Learning Representations*, volume 2024, pp. 2369–2396, 2024.

Ramanan Sekar, Oleh Rybkin, Kostas Daniilidis, Pieter Abbeel, Danijar Hafner, and Deepak Pathak. Planning to explore via self-supervised world models. In *International conference on machine learning*, pp. 8583–8592. PMLR, 2020.

Trevor Standley, Amir Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? In *International conference on machine learning*, pp. 9120–9132. PMLR, 2020.

Haixin Wang, Hejie Cui, Chenwei Zhang, Xin Liu, Shuowei Jin, Shijie Geng, Xinyang Zhang, Nasser Zalmout, Zhenyu Shi, and Yizhou Sun. T ^{2} po: Uncertainty-guided exploration control for stable multi-turn agentic reinforcement learning. *arXiv preprint arXiv:2605.02178*, 2026.

Zirui Wang, Yulia Tsvetkov, Orhan Firat, and Yuan Cao. Gradient vaccine: Investigating and improving multi-task optimization in massively multilingual models. *arXiv preprint arXiv:2010.05874*, 2020.

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, 2025.

Zhiheng Xi, Yiwen Ding, Wenxiang Chen, Boyang Hong, Honglin Guo, Junzhe Wang, Xin Guo, Dingwen Yang, Chenyang Liao, Wei He, et al. Agentgym: Evaluating and training large language model-based agents across diverse environments. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27914–27961, 2025.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh J Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Advances in Neural Information Processing Systems*, 37:52040–52094, 2024.

John Yang, Carlos Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. Swe-agent: Agent-computer interfaces enable automated software engineering. *Advances in Neural Information Processing Systems*, 37:50528–50652, 2024.

Dengjia Zhang, Xiaoou Liu, Lu Cheng, Yaqing Wang, Kenton Murray, and Hua Wei. Selaur: Self evolving llm agent via uncertainty-aware rewards. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 424–436. Springer, 2026.

Jenny Zhang, Joel Lehman, Kenneth Stanley, and Jeff Clune. Omni: Open-endedness via models of human notions of interestingness. In *International Conference on Learning Representations*, volume 2024, pp. 6074–6120, 2024.

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.