RE-0: VERIFIED RECURSIVE IMPROVEMENT OF EMBODIED CODE-AS-POLICY AGENTS THROUGH LOCAL ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Code-as-Policy agents accomplish long-horizon embodied tasks by generating and executing code, yet continually improving them with teachers that are stronger but not globally reliable remains a key challenge. Existing distillation methods typically treat the teacher's complete behavior as the supervision target and thus misassign training credit on states where the teacher fails. We propose **RE-0**, a recursively verified policy improvement framework: rather than assuming that the teacher globally outperforms the student, RE-0 requests local corrections from the teacher on the student's own failure histories and checks in the environment whether each correction is genuinely beneficial; verified corrections yield immediate improvement. Building on this, we propose **RE-OPD**, which turns verified interventions into supervision for on-policy distillation. Only counterfactually verified teacher interventions provide distribution-level supervision, weighted by their measured local benefit, and the improvement they induce is projected back into the standalone student—so both where supervision is applied and how much credit the teacher receives co-evolve with the student policy. We further prove that the student's per-round gain is lower-bounded by its verified intervention gain up to verification and projection error terms. Experiments across multiple Code-as-Policy embodied tasks show that RE-0 improves both teacher-assisted execution and the standalone student, and generalizes to novel robots and scenes.

1 INTRODUCTION

Code-as-Policy agents turn language models into policies that interact with an environment through executable programs—code as the interface for compositional control, tool calling, and long-horizon reasoning. Policy errors are then strongly history-dependent: one harmful segment not only fails the current action but alters the environment state, the program namespace, and the subsequent observable history. Improving such an agent therefore poses a question more basic than *what code should be generated*: *where, along the execution history actually induced by the student, should corrective credit be assigned.*

Existing approaches improve code policies through expert-trajectory supervision, outcome-level reinforcement learning, or teacher distillation, but none answers the question that matters in long-horizon embodied tasks: whether one concrete teacher intervention on a real student failure produced a positive causal contribution. Even a strong teacher accumulates errors through perception ambiguity or earlier executions, while a teacher unable to finish the task alone may still decide better at a local failure boundary the student has reached; neither global success rate nor prefix-level reliability determines whether a teacher action deserves training credit.

We therefore reformulate teacher supervision as *local counterfactual credit on student-induced histories*: on a history h genuinely visited by the current student, does a bounded teacher correction beat the student's action, and does the advantage persist after control returns to the same student? Three coupled challenges arise: localizing the *earliest* plausibly causal harmful boundary; comparing correction and student from the *identical* environment state; and absorbing verified assistance into the student so deployment needs no teacher. We call this **verifier-mediated local teacher credit assignment**.

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.

Our framework, **RE-0** (Figure 1), solves it recursively. Given a failed trajectory of the current student π_k, a diagnostic agent localizes the earliest candidate harmful boundary and constructs repair contracts from the smallest scope; the teacher emits one bounded code macro-action within the contract, never taking over the remaining trajectory. RE-0 restores the environment to the snapshot saved at that boundary, executes the student baseline and the teacher correction under shared randomness, and admits the intervention only if the lower confidence bound of the paired advantage is positive; control then returns to the student, yielding a verified local assistant μ_k that intervenes only where the teacher has positive local credit. To convert this transient advantage into standalone capability, **RE-OPD** distills on the histories actually visited by μ_k: the current student is the retention term, the contract-constrained teacher the local target, and the distillation weight is the verified counterfactual advantage; the next student then regenerates trajectories and supervision locations are re-localized on the new occupancy, giving the recursion $\pi_k \to \mu_k \to \pi_{k+1}$ whose per-round gain is lower-bounded by the verified intervention gain up to verification and projection error, with no global teacher dominance required. Our ablations locate the load-bearing component precisely: it is the verified admission of patches (the corpus), not the specific weighting—uniform SFT on the verified corpus is within trainer-seed variance of the weighted objective, while unverified corpora fail (composition and selection analyses in §4). On six long-horizon tasks and a physical robot platform, RE-0 lifts the base from 4–68% to 62–100% with 3–67 verified rows (details in §4).

Our contributions are: (1) verifier-mediated local teacher credit on student-induced histories; (2) **RE-OPD**, which recursively projects verified local improvement into a teacher-free student with a finite-sample improvement guarantee; and (3) empirical evidence across six long-horizon tasks and physical deployment that verified corpus selection—not weighting alone—is the load-bearing mechanism.

2 Related Work

End-to-end vision-language-action models predict actions directly; the code-as-policy interface instead keeps an explicit, compositional program between language and control—precisely what makes event-level teacher patches and environment verification well defined.

Code-as-Policy and embodied code agents. Large language models have been used to map natural-language tasks into embodied plans, callable skills, and executable robot programs: SayCan constrains planning with environment affordances, Inner Monologue folds environment feedback into closed-loop planning, and Code as Policies and ProgPrompt exploit code structure, control APIs, and program context to generate embodied policies (Ahn et al., 2022; Huang et al., 2022; Liang et al., 2023; Singh et al., 2023). Recent work repairs code policies from vision and execution traces, or searches and evolves high-return programs with rollout evaluators (Lin et al., 2025; Sygkounas et al., 2026). These methods focus on inference-time planning, program repair, or candidate search; RE-0 asks how corrections discovered *during execution* become verifiable teacher credit amortized into a teacher-free student.

On-policy distillation and interactive imitation. Interactive imitation queries an expert on learner-induced states to mitigate covariate shift, with DAgger as the classic form (Ross et al., 2011); in autoregressive models, GKD distills on student-generated sequences and DistiLLM lowers its cost via adaptive off-policy reuse (Agarwal et al., 2024; Ko et al., 2024). For multi-turn agents, prefixes near the student occupancy can themselves make the teacher target unreliable, motivating reliability-aware replay approximations of OPD. RE-0 shares the goal of learning on the student distribution but never presumes teacher reliability on queried histories: it estimates the teacher's local counterfactual advantage against student behavior from the same simulation snapshot and projects only environment-verified positive-credit interventions.

Agentic correction, program repair, and verifiers. Self-Refine, Reflexion, and CRITIC revise generations with self-feedback, verbal memory, or external tools (Madaan et al., 2023; Shinn et al., 2023; Gou et al., 2024); in code generation, CodeRL scores programs from compilation and unit tests (Le et al., 2022). Such methods typically regenerate whole answers, select among candidates at the outcome level, or feed execution traces back as prompts. RE-0 instead localizes the *earliest* harmful code boundary in the student's trajectory, compares the student baseline and the teacher correction

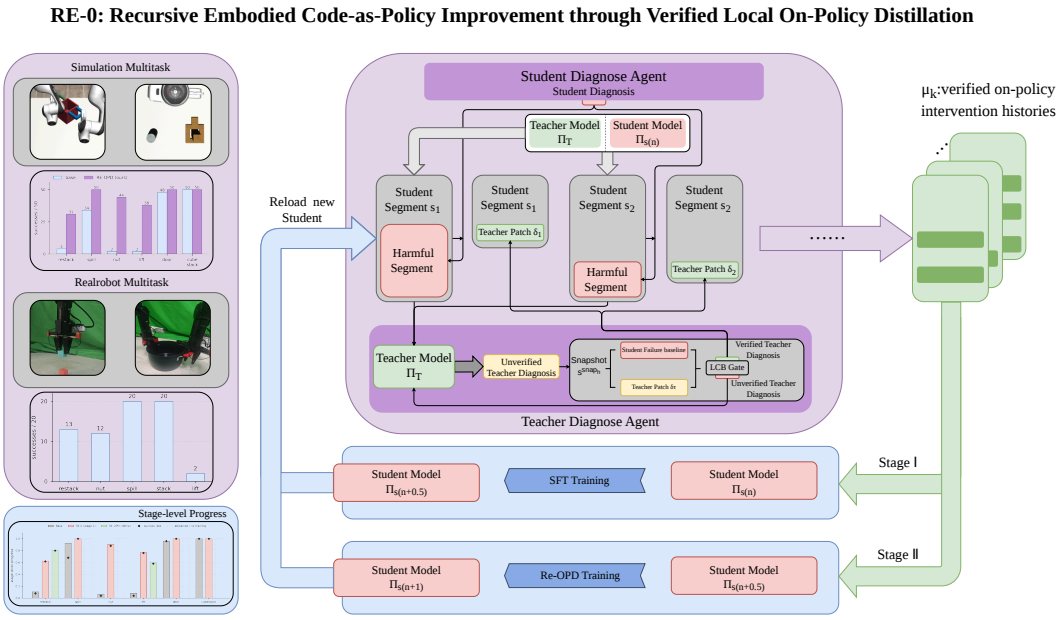

Figure 1: **RE-0 overview.** RE-0 localizes harmful segments in the current student's trajectories, requests bounded teacher repairs, and verifies them by paired counterfactual rollout from the same snapshot (LCB gate). Verified interventions μ_n are distilled into the standalone student via cold-start SFT (Stage I) and RE-OPD refinement (Stage II); the student then regenerates trajectories, closing the loop. **Left:** simulation, real-robot, and stage-level results.

from the *identical* environment snapshot, and confines teacher control to a minimal verified macro-action after which the student regenerates the suffix; the verifier thus decides not only whether a candidate succeeds but whether the teacher's distribution earns local training credit.

Selective supervision and local credit assignment. Prior work selectively queries experts or weights demonstrations by uncertainty or quality (Zhang & Cho, 2017; Kelly et al., 2019; Zhang et al., 2021). RE-0 instead defines the credit object itself: the teacher's conditional behavior on a specific student history, evaluated by paired same-state counterfactual advantage.

3 RE-0: Recursive Verified Policy Improvement

Learning from a locally better but globally imperfect teacher. Standard on-policy distillation matches the teacher's conditional distribution on histories visited by the student. RE-0 assumes no global policy dominance between student and teacher: the teacher holds positive *local* counterfactual advantage only on some student-induced histories, and supervision credit is assigned not to teacher trajectories as wholes but to the marginal improvement an intervention produces over the student's own behavior (Eq. 1). The correction δ takes over only a minimal executable segment and immediately returns control to the current student π_k. A diagnostic agent localizes failure boundaries that may carry positive credit, an auditable counterfactual executor verifies the credit is real, and RE-OPD amortizes the verified improvement into the next independent student—verifier-mediated local credit assignment.

3.1 Local Teacher Credit on Student-Induced Histories

With $J(\pi)$ the expected discounted return, the policy-visible history at the t-th code decision point is $h_t = (o_0, c_0, \ldots, o_t)$, bounded code segments acting as macro-actions, $d^t_{\pi_k}$ the step-t history distribution of the generation-k student, and s^{snap}_t the verifier's policy-invisible snapshot. RE-0 asks whether the teacher decides better *locally* on histories actually induced by the current student: for the teacher's local correction policy $\tau_k(\delta \mid x, h_t)$, whose corrections return control to π_k after

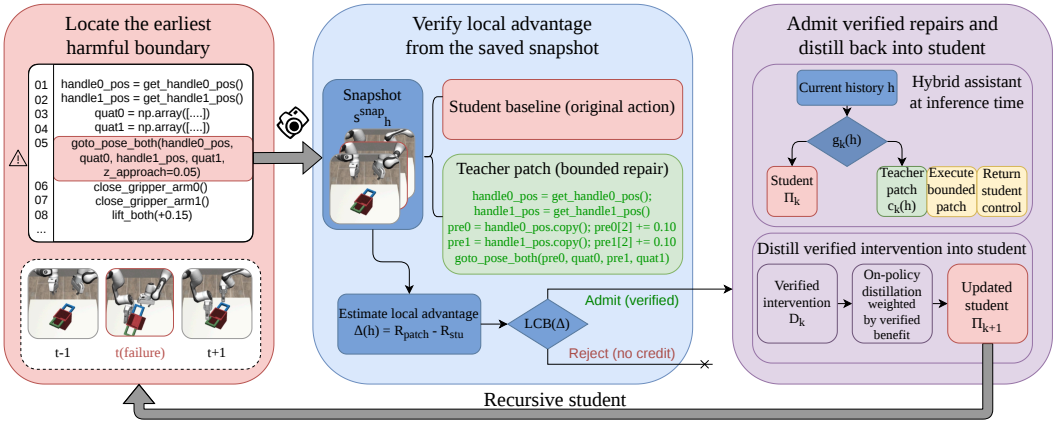

Figure 2: **One recursion on a failing two-arm lift program. Locate:** the earliest harmful boundary t(failure). **Verify:** from the same snapshot, student baseline vs. teacher patch under shared randomness; admitted iff $\underline{\Delta} > 0$. **Distill:** verified repairs are credit-weighted into the standalone student; deployment uses no teacher calls.

executing, the expected local credit is (formal definitions in Appendix B)

$$\Delta_k(h_t) = \mathbb{E}_{\delta \sim \tau_k(\cdot|x,h_t)} \left[A_t^{\pi_k}(h_t, \delta) \right]. \tag{1}$$

Equation 1 captures the counterfactual marginal contribution of teacher intervention relative to the current student. If $\Delta_k(h_t) > 0$, the teacher's local policy is better than continuing the student at h_t; if $\Delta_k(h_t) \leq 0$, the teacher's supervision on that history must not be accepted automatically, regardless of its overall strength.

The ideal allocation—supervision only on $\mathcal{H}_k^+ = \{(t, h_t) : \Delta_k(h_t) > 0\}$ under budget B, a fractional-knapsack LP (appendix)—is not directly observable: $\Delta_k(h)$ is hidden and the history space forbids exhaustive verification. RE-0 therefore uses diagnostics to localize executable failure boundaries that may carry positive local credit, then decides by counterfactual execution from identical states.

3.2 Verified Local Assistance

Equation 1 defines the teacher's true local credit, which cannot be observed directly. RE-0 approximates this ideal allocation with a two-stage mechanism: a diagnostic agent localizes the executable failure boundary that may carry positive local credit, and an environment verifier decides through same-state counterfactual execution whether the intervention yields a real gain. The former shrinks the candidate space; the latter decides whether teacher supervision enters the hybrid policy.

Given a student-executed trajectory ζ, the student diagnostic agent outputs the earliest candidate failure history with structured evidence, $(\widehat{h}, z)$: z collects runtime exceptions, unrealized expected effects, or other structured failure evidence, and the execution system saves the full simulation snapshot $s^{\text{snap}}_{\widehat{h}}$; if no executable, causally consistent boundary is found, no teacher call fires. A teacher diagnostic then judges whether the failure is locally repairable and emits repair contracts of monotonically increasing scope cost κ (formally, $(\mathfrak{X}(\widehat{h}, z), g_k^{\text{diag}}(\widehat{h})) = \mathcal{A}_k^T(x, \widehat{h}, z)$; definitions in the appendix). Each contract constrains the repair boundary, the allowed program symbols and APIs, and the maximum segment length. RE-0 verifies contracts in increasing $\kappa(\chi)$ and selects the smallest *verified* repair scope—*minimal necessary repair* means the smallest verified scope, not the shortest candidate string (selection rule and validity predicate in the appendix). If $g_k^{\text{diag}}(\widehat{h}) = 0$ or no contract qualifies, no intervention fires; we write the selected contract teacher distribution as $\bar{\tau}_k^{\chi_k^\star}$.

For each teacher candidate, RE-0 restores the environment to $s^{\text{snap}}_{\widehat{h}}$. Under shared subsequent randomness $\xi_1, \ldots, \xi_M$, the teacher correction and a student resample $c_m^S \sim \pi_k(\cdot \mid x, \widehat{h})$ are executed and control returns to the same student π_k after the segment completes; the paired counterfactual advantage estimate $\widehat{A}_k(\widehat{h}, \delta)$ is the M-average of the return difference $(\delta \to \pi_k)$ minus

$(c_m^S \to \pi_k)$. RE-0 verifies the *expected* credit of the contract teacher distribution: averaging K candidates $\delta_j \overset{\text{i.i.d.}}{\sim} \bar{\tau}_k(\cdot \mid x, \widehat{h})$ with a two-layer confidence radius controlling both teacher-candidate and rollout sampling error,

$$\widehat{\Delta}_k(\widehat{h}) = \tfrac{1}{K}\sum_j \widehat{A}_k(\widehat{h}, \delta_j), \qquad \underline{\Delta}_k(\widehat{h}) = \widehat{\Delta}_k(\widehat{h}) - b_k(\widehat{h}; K, M, \alpha). \tag{2}$$

When verifying multiple histories we allocate $\alpha_{k,h}$ with $\sum_{k,h}\alpha_{k,h} \le \alpha$ for simultaneous coverage over the whole recursion; concentration bounds appear in the appendix. The environment gate is $g_k^{\text{env}}(\widehat{h}) = \mathbf{1}[\underline{\Delta}_k(\widehat{h}) > \epsilon]\cdot\mathbf{1}[\text{Valid}(\chi_k^\star, \bar{\tau}_k, s_{\widehat{h}}^{\text{snap}})]$, with $\epsilon \ge 0$ the minimum advantage threshold and Valid the execution-safety, program-context, and snapshot-identity predicate (appendix); candidate returns never post-select a single best candidate.

Combining the budget allocation q_k (appendix), the final intervention probability is $g_k(h) = q_k(h)\, g_k^{\text{diag}}(h)\, g_k^{\text{env}}(h)$, giving the verified local hybrid policy

$$\mu_k(c \mid x, h) = \big(1 - g_k(h)\big)\,\pi_k(c \mid x, h) + g_k(h)\,\bar{\tau}_k(c \mid x, h). \tag{3}$$

From here on, the teacher policy in Eq. 1 is taken to be the same contract distribution $\bar{\tau}_k$. The teacher takes over only the current bounded repair segment; after it executes, the student regenerates subsequent code from the new observation rather than executing a suffix generated by the old policy.

Remark (rewind semantics). The pipeline runs two passes—a live attempt that localizes candidate boundaries, then verification from the stored snapshot—but both arms of every paired comparison start from the *same* snapshot, so the first pass is a common data-collection cost that cancels in Eq. 4; μ_k is equivalently a one-pass policy of the *augmented* process whose state includes the snapshot stack and the first-pass diagnostic evidence, and the bound applies to the post-rewind continuation return of that process, not to a snapshot-free online policy (appendix).

By the history–state performance-difference lemma, $J(\mu_k) - J(\pi_k) = \sum_t \gamma^t \mathbb{E}_{h\sim d^t_{\mu_k}}[g_k(h)\Delta_k(h)]$. Define the verifiable intervention gain $\widehat{\mathcal{G}}_k = \sum_t \gamma^t \mathbb{E}_{h\sim d^t_{\mu_k}}[g_k(h)[\widehat{\Delta}_k(h)]_+]$ and the verification error budget $\mathcal{E}_{\text{verify},k} = \sum_t \gamma^t \mathbb{E}_{h\sim d^t_{\mu_k}}[g_k(h) b_k(h)] + \mathcal{E}_{\text{sys},k}$, where $\mathcal{E}_{\text{sys},k}$ collects system approximation errors such as snapshot restoration and execution identity. Whenever returns are bounded, contract supports are valid, and the simultaneous-coverage condition of Eq. 2 holds, with probability at least $1-\alpha$,

$$J(\mu_k) - J(\pi_k) \;\ge\; \widehat{\mathcal{G}}_k - \mathcal{E}_{\text{verify},k}. \tag{4}$$

Thus RE-0 converts verified positive local credit into expected performance gain of the hybrid policy without requiring global teacher dominance.

3.3 RE-OPD: Amortizing Local Credit into the Student

The verified local hybrid μ_k obtains higher expected return through local teacher interventions, but executing it still requires teacher calls and environment verification. RE-OPD amortizes these transient local advantages into student parameters, so that π_{k+1} reproduces the improved behavior without teacher access.

Recall the hybrid of Eq. 3. The ideal projection finds the independent student closest to μ_k on the histories μ_k actually visits, $\pi_{k+1} = \arg\min_{\pi_\theta} \sum_t \gamma^t \mathbb{E}_{h\sim d^t_{\mu_k}}[D_{\text{KL}}(\mu_k(\cdot \mid x, h) \| \pi_\theta(\cdot \mid x, h))]$. Training histories cover both the failure states the student would visit and the new states induced by the student after teacher correction—not fixed teacher trajectories, nor mechanically stitched student suffixes.

Because μ_k is itself the g_k-mixture of π_k and $\bar{\tau}_k$ (Eq. 3), substituting into the KL and dropping parameter-independent entropy terms turns the ideal projection *exactly* into a g_k-weighted sum of a student-retention term and a teacher-projection term; RE-OPD re-weights teacher supervision by verified local credit:

$$\mathcal{L}_{\text{RE-OPD}}(\theta) = \sum_t \gamma^t\, \mathbb{E}_{h\sim d^t_{\mu_k}}\Big[\big(1 - g_k(h)\big)\, D_{\text{KL}}\big(\pi_k \,\|\, \pi_\theta\big) + \lambda\, g_k(h)\, w_k(h)\, D_{\text{KL}}\big(\bar{\tau}_k \,\|\, \pi_\theta\big)\Big], \tag{5}$$

with all conditional distributions conditioned on (x, h), λ controlling local distillation strength, and the local credit weight $w_k(h) = [\underline{\Delta}_k(h)]_+$. Equation 5 is a credit-weighted surrogate inspired by

the exact mixture projection, not a strict upper bound of the ideal projection for arbitrary $\lambda w_k(h)$; through $g_k(h)w_k(h)$ it decides *how much* teacher supervision earns training credit *where*.

Restricted estimators implement the same projection (derivation in the appendix): Top-K token probabilities give a truncated sparse estimate, and a black-box teacher degenerates to the trajectory-level hard estimator $\ell_{\mathrm{hard}}(h, \delta) = -\lambda g_k(h)w_k(h) \log \pi_\theta(\delta \mid x, h)$ used throughout our experiments.

Optimizing Eq. 5 controls the occupancy-weighted projection error $\varepsilon_{\mathrm{proj},k}$ (definition in the appendix), for which $[J(\mu_k) - J(\pi_{k+1})]_+ \le C_{T,\gamma,R}\sqrt{\varepsilon_{\mathrm{proj},k}}$ with $C_{T,\gamma,R}$ depending only on horizon, discount, and return bound (proof in the appendix); combining this with Eq. 4 yields the core result.

Theorem 1 (Recursive verified policy improvement)**.** *If returns are bounded, contracts are valid, the counterfactual verifier satisfies the simultaneous-coverage condition of Eq. 2, and the occupancy-weighted projection error of RE-OPD is bounded, then with probability at least* $1 - \alpha$*,*

$$J(\pi_{k+1}) - J(\pi_k) \;\ge\; \widehat{\mathcal{G}}_k - \mathcal{E}_{\mathrm{verify},k} - C_{T,\gamma,R}\sqrt{\varepsilon_{\mathrm{proj},k}}. \tag{6}$$

Hence $J(\pi_{k+1}) > J(\pi_k)$ *whenever* $\widehat{\mathcal{G}}_k > \mathcal{E}_{\mathrm{verify},k} + C_{T,\gamma,R}\sqrt{\varepsilon_{\mathrm{proj},k}}$.

After each projection, RE-0 regenerates trajectories with π_{k+1} from the raw task input and re-allocates local teacher credit on the new occupancy.

Remark 1 (Scope of the guarantee)**.** *$\widehat{\mathcal{G}}_k$ and $\mathcal{E}_{\mathrm{verify},k}$ are measured per event by the audit and the admission rule; the projection term is not estimated a priori but guarded by the outer acceptance criterion (Remark 2), which retains a generation only if its measured performance does not regress—verified at the credit level, guarded by measurement at the policy level (per-round sign accounting in Appendix D).*

3.4 Cold Start and Recursive Refinement

The construction assumes nothing about *who* occupies the policy seat: replacing d_{π_k} by d_σ for any seat policy σ defines $\mu^\sigma = \mathcal{I}(\sigma)$, and under the simultaneous coverage of Eq. 2 the guarantee of Eq. 4 transfers verbatim, with $\mu^\sigma = \sigma$ when the gate is identically zero—no global capability relation, no training (Proposition 1, appendix).

RE-OPD-cold. Taking $\sigma = \tau$ puts the teacher in the seat: its own failure histories are likewise diagnosed, repaired under progressive contracts, and verified by the same gate, and the verified-completed trajectories form the cold corpus $\mathcal{D}_0$ on which full-trajectory supervision yields π_1. $\mathcal{D}_0$'s quality is guaranteed by Proposition 1, not by global teacher reliability—even a teacher unable to complete the task contributes trajectories inheriting verified self-improvement credit, and π_1 inherits the lower bound of Theorem 1 with the teacher in the seat (Corollary 1, appendix; the cold surrogate is completion-conditioned there).

RE-OPD-refine. Later generations put the student in the seat, $\pi_{k+1} = \mathcal{P}_\Theta(\mathcal{I}(\pi_k))$; the projections from the teacher and student seats are **RE-OPD-cold** and **RE-OPD-refine**, sharing all machinery and differing only in the occupancy source of the corpus. Termination is structural: if $\underline{\Delta}_k(h) \le \epsilon$ for almost all $h \in \mathrm{supp}(d_{\pi_k})$, the gate closes, Eq. 5 degenerates to the retention term, and the recursion safely abstains (Corollary 2, appendix)—triggered by exhausted headroom or by teacher effect below the verification noise floor b_k.

Remark 2 (Deployment criterion and two conditions for useful refinement)**.** *Given threshold ρ, halt if $\widehat{J}(\pi_1) \ge \rho$ and otherwise refine, re-checking after each round. Refinement is useful only when* headroom *η (student below ρ) and* verifiable teacher effect *ν (admitted credit mass above the noise floor) hold* jointly*; either alone is insufficient, as the lift experiment in §4.2 shows empirically.*

4 Experiments

4.1 Experimental Setup

Tasks and environments. We evaluate on six long-horizon robosuite Code-as-Policy tasks—*cube restack*, *spill wipe*, *nut assembly*, *two-arm lift*, *cube stack*, and *door opening*—covering single/dual

Task	Base	Flash-API	RE-OPD-cold (n rows)	RE-OPD-refine
restack	4/50[a]	2	**31** (3)	**40**[b]
spill	34	33	**50** (67)	—[c]
nut	2	26	**44** (27)	—
lift	2	6	**38** (7)	29[d]
door	48	—	**50** (40)	—
cubestack	50	50	50 (0, no-train)[e]	—

Table 1: **Main results** (successes / 50 held-out seeds). Flash-API: same teacher, single-shot without diagnose–repair–verify. n: verified training rows. "—": criterion abstained, not unmeasured. Trainer-seed replication (three seeds, Appendix D): restack 31/44/36, spill 50/33/33, lift 38/33/35, nut 44/40/43, door 50/50/50.

arms, rigid/articulated objects, and coarse/fine manipulation (details in the appendix). Base-model success spans 4% to 100% (restack 8%, lift 4%, nut 4%, spill 68%, door 96%, cubestack 100%), a full difficulty spectrum; each task keeps a held-out seed set disjoint from training collection.

Models, seats, collection, and training. The student is Qwen3.8-27B (Qwen Team, 2026) fine-tuned with LoRA; the teacher and both diagnostic agents are the qwen3.8-max API. The teacher is black-box (no conditional distributions), so all projections use the trajectory-level hard estimator; Eq. 5 thus becomes a supervised loss weighted by verified credit, with the same optimizer family as SFT—the difference is not the loss form but *what enters it*: supervision locations, per-event weights $g_k w_k$, and the retention term $(1 - g_k)$ are decided by environment verification, not teacher sampling. Baselines (plain SFT, RFT/STaR, GRPO, OPD without verification) share the optimizer, loss family, and evaluation protocol, differing only in corpus provenance and quality assurance. Collection sends fresh-rollout failures (cycles the student already solves are skipped) through the mechanism of §3.2 with LCB admission per Eq. 2; the same pipeline on the teacher seat and the student seat yields the **RE-OPD-cold** and **RE-OPD-refine** corpora (hyperparameters in the appendix). *Abstention rule*: tasks already at $\geq$80% success do not enter refinement—the practical instance of Corollary 2—and are marked as abstained. The execution system saves full simulation snapshots at code decision points; diagnostic evidence z includes 1–2 perception-derived physical quantities per task (§4.4).

Evaluation protocol. All methods use 50 paired held-out seeds per task (Wilson 95% CIs in Appendix D; paired per-seed outcomes for McNemar tests ship with the released artifacts) and share identical training and evaluation infrastructure apart from the variable under study. One protocol caveat: stage-entry (ρ) decisions reused the 50-seed evaluation cohort; collection seeds are always disjoint, and re-evaluation of all base and cold checkpoints on a disjoint, never-used 50-seed cohort reproduces the reported conclusions within binomial variation (Appendix D).

4.2 Main Results

Table 1 reports held-out success on 50 seeds across the six tasks.

Sample efficiency of verified data. RE-OPD-cold lifts the base from 4–68% to 62–100% with 3–67 verified rows (restack 62% with 3 rows, lift 76% with 7, nut 88% with 27)—improvement is not monotone in row count; the decisive factor is the verified credit each row carries (Corollary 1), not corpus size. The same teacher's single-shot calls reach only 4% on restack while solving nut (52%): without the diagnose–repair–verify loop, teacher capability does not land on hard tasks. *When no teacher is needed* is decided by the framework itself: cubestack is saturated and exempted; door goes 96%$\rightarrow$100% with 40 rows.

Adaptive criterion. Refinement happens only where the target is unmet: restack 62%$<\rho$=80% refines to 80%; spill is already at 100%, its admission is empty, and the framework correctly abstains—a runtime instance of Corollary 2. Admissions decay with occupancy (6/6 at the π_1 round vs. 4/15 at π_2; Figure 3b; per-round accounting in the appendix). The lift row exercises the outer acceptance test (Remark 1): 76%$<\rho$ but π_2 admits one event (w=0.069)—thin verified credit; the projection regresses to 58% with branch collapse (Appendix D, note (d); the regression replicates at

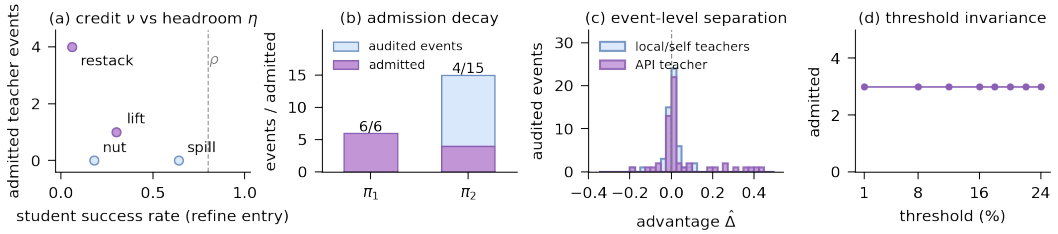

Figure 3: **Credit analysis.** (a) Headroom alone does not imply credit: all tasks sit below ρ=0.80 yet admissions range 0–4. (b) Admission decay across rounds. (c) $\widehat{\Delta}$ separation: admitted vs. abstained over 110 events. (d) Threshold sweep: the admitted set is invariant.

training seed 2, 76%→46%), and the criterion halts the recursion, disclosing the regressed checkpoint rather than propagating it: η without ν yields a harmful projection, and it is the stage-entry measurement that catches it.

Failure of pooling. Naively pooling verified data across tasks causes severe interference despite larger corpus size (Table 8, appendix), consistent with the allocation failure mode in §3.1.

4.3 Verification and Credit Analysis

The admission ledger measures teacher effect. The same gate, applied to different teachers, yields a monotone ladder (appendix, Table 6): 0 admissions for student self-play (34 cycles) and for narrow local experts (32 events; ∼10% rescue ceiling, deterministic contract-format failures), versus 4 admissions on 13 events for the strong API teacher (rescue rates 19–47%). These gradings precede any training: the verifier directly measures the teacher effect ν of Remark 2—not "who is stronger," but "who holds verifiable local credit on the student's failure history."

Strong separation makes the strict gate free; the gate is two-sided. Admitted events cluster at $\widehat{\Delta} \in [0.19, 0.44]$ while abstained ones satisfy $|\widehat{\Delta}| \leq 0.04$ (Figure 3c); sweeping the threshold from 0.24 to 0.01 leaves the admitted set unchanged (Figure 3d)—the LCB gate's statistical strictness costs no sensitivity. The verifier also rejects harm: one API patch dropped the baseline 8/16→1/16 ($\widehat{\Delta} = -0.44$) and was rejected. Credit assignment is a decision on the causal *sign* of an intervention, not one-directional imitation.

Passing the gate ≠ policy flip. Weight-scale and normalization ablations cannot recover the gain of the verified corpus and can induce catastrophic forgetting (Appendix E), showing that verification determines corpus quality while weighting alone cannot substitute for composition-preserving training. This isolates the load-bearing variable: the gate's product is the corpus.

GRPO control. With terminal rewards in the same environments, GRPO (Shao et al., 2024) undergoes entropy collapse (0.30→0.002) and transfers none of it: 0–1/50 on the non-saturated tasks against base 8/68/4%, retaining only the saturated task (details and tuning notes in the appendix). Sparse terminal rewards cannot localize the harmful decision inside a long program, in mechanistic contrast to locate–verify–weight.

4.4 Ablations

Selection-rule ablation: the gate is the source of credit quality. Holding the candidate journal and training recipe fixed, LCB admission outperforms unfiltered and sign-only selection on lift across three trainer seeds (Table 2; Appendix E).

Corpus composition is decisive. On restack, success-filtered teacher demonstrations remain at base (3/50), whereas three verified event patches reach 46/50. Outcome filtering selects the identical corpus on this task, so the discriminating admission-rule comparison is Table 2 (Appendix E).

Selection rule (lift journal)	Rows	Successes / 50			Mean
		s_1	s_2	s_3	
None (all candidates)	25	6	2	3	3.7
Sign ($\widehat{\Delta} > 0$)	7	5	4	4	4.3
LCB (paired, Eq. 2)	3	**11**	**5**	**13**	**9.7**
Outcome (teacher programs)	27	5	9	4	6.0

Table 2: **Selection-rule ablation** (lift; three trainer seeds; uniform weights). LCB dominates unfiltered and sign rules in every seed pairing.

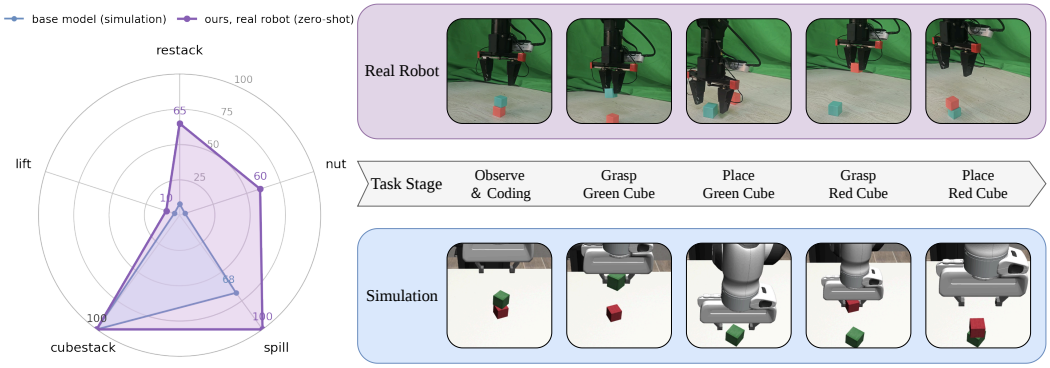

Figure 4: **Zero-shot sim-to-real transfer.** Real-robot execution of the simulation-trained policies on the dual-arm AgileX PiPER platform: task keyframes (left), hardware success counts on 20 trials per task (middle), and hardware success rates against the simulation base model (right). The deployed policy code is identical to the simulation policy—no hardware-specific edits.

Objective-composition quartet; teacher necessity. Replacing only the objective composition (restack; separate 50-seed cohort; armwise scores, definitions, and per-cohort variance in the appendix) confirms the decomposition of Eq. 5: stripping the student prefix (patch-only) or the retention term (weighted) collapses the gain that the full objective and uniform credit largely retain. Student self-bootstrap yields 0 admissions on four tasks, echoing §4.3.

4.5 Generalization and Real-Robot Deployment

Transfer and deployment. Cross-task evaluation shows selective rather than universal transfer: restack training improves nut while spill's wiping dynamics do not carry, and pooled multi-task training exhibits substantial interference (Appendix F). We additionally deploy the distilled, teacher-free policies zero-shot, without hardware-specific policy edits, on a dual-arm AgileX PiPER platform with SAM3-based open-vocabulary perception, obtaining 13/20 restack, 12/20 nut, 20/20 spill, 20/20 cubestack, and 2/20 lift successes (Figure 4, Appendix F). Verification is used only during simulation-side collection; hardware deployment makes no teacher calls.

5 Conclusion

We recast self-improvement of embodied Code-as-Policy agents as a **locate–verify–weight** recursion: teacher interventions are minimal patches, verified event-wise by same-snapshot counterfactual experiments, admitted only with finite-sample confidence, and returned to the student after a single bounded macro-action. Across six tasks and physical deployment, 3–67 verified rows lift the base from 4–68% to 62–100%, while the same admission ledger doubles as a measurement instrument driving self-termination. The central empirical finding is that verified corpus selection, not weighting alone, is the load-bearing mechanism—confirmed by selection ablations across three trainer seeds (Table 2); making the ideal allocation of supervision under composition constraints explicit is the natural next step, with the accumulated admission ledger as its measurement instrument and every number in it traceable to a logged counterfactual experiment.

AI USE STATEMENT

Roles of AI systems. All policies evaluated in this work are LLM-based code-as-policy agents; they constitute the *object of study* and are reported as experimental setup, not as AI assistance.

Required disclosure. We used generative AI tools to: (i) design and provide feedback on research methodology and experiments (experiment prioritization and ablation design were discussed with an AI assistant; all protocol parameters were fixed in advance by the authors and never altered); (ii) implement methods (operational scripts for training, evaluation, and data assembly were drafted with AI assistance and executed under author supervision; the core environment and pipeline code is the authors' own); (iii) interpret results (log analysis and attribution of negative results were AI-assisted and re-verified by the authors against raw artifacts); and (iv) assist with translation of the manuscript draft. We did *not* use generative AI to generate synthetic datasets, to formulate or prove mathematical claims, or to produce any experimental data: every number in this paper is programmatically extracted from logged evaluation runs of the described pipeline.

Recommended disclosure. We additionally used generative AI tools to draft parts of the paper, edit text for readability, create scientific figures (all figure scripts and underlying event data will be released), suggest paper structure, and format references.

Responsibility. All AI-assisted code was executed and tested by the authors, all reported numbers were cross-checked against raw logs, and all AI-assisted text was reviewed and revised by the authors. We take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We will release the complete pipeline and all artifacts: (i) environment configurations for all six robosuite tasks; (ii) the full admission journals—110 audited events with paired counterfactual advantages, lower confidence bounds, and assigned training weights; (iii) every verified training parquet with row-level provenance to admission events; (iv) raw evaluation JSONs underlying all reported evaluation tables and ablations, together with the admission journals underlying the analysis figures; and (v) training configurations. All numbers in the paper are traceable to these artifacts via a release manifest.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, et al. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations (ICLR)*, 2024.

Michael Ahn, Anthony Brohan, Noah Brown, et al. Do as i can, not as i say: Grounding language in robotic affordances. In *Conference on Robot Learning (CoRL)*, 2022.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. Critic: Large language models can self-correct with tool-interactive critiquing. In *International Conference on Learning Representations (ICLR)*, 2024.

Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, et al. Inner monologue: Embodied reasoning through planning with language models. In *Conference on Robot Learning (CoRL)*, 2022.

Michael Kelly, Chelsea Sidrane, Katherine Driggs-Campbell, and Mykel J. Kochenderfer. Hg-dagger: Interactive imitation learning with human experts. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. Distillm: Towards streamlined distillation for large language models. In *International Conference on Machine Learning (ICML)*, 2024.

Hung Le, Yue Wang, Akhilesh Deepak Gotmare, Silvio Savarese, and Steven C. H. Hoi. Coderl: Mastering code generation through pretrained models and deep reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

Zefu Lin et al. Embodiedcoder: Parameterized embodied mobile manipulation via modern coding model. *arXiv preprint arXiv:2510.06207*, 2025.

Aman Madaan, Niket Tandon, Prakhar Gupta, et al. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Qwen Team. Qwen3.8-27b model card. `https://huggingface.co/Qwen/Qwen3.8-27B`, 2026.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

Zhihong Shao, Peiyi Wang, Qihao Zhu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, et al. Progprompt: Generating situated robot task plans using large language models. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

Alkis Sygkounas, Victor Aregbede, Amy Loutfi, and Andreas Persson. Memento: Memory-guided memetic code-as-policy evolution. *arXiv preprint arXiv:2607.22832*, 2026.

Jiakai Zhang and Kyunghyun Cho. Query-efficient imitation learning for end-to-end simulated driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2017.

Songyuan Zhang, Zhangjie Cao, Dorsa Sadigh, and Yanan Sui. Confidence-aware imitation learning from demonstrations with varying optimality. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

A LIMITATIONS

(1) The allocation layer is not closed: $w \equiv \mathrm{LCB}$ suppresses patches to 1–1.5% weight, $\Sigma w \to 1$ normalization fails to flip and destroys composition, and unprotected distillation forgets; converting verified credit into policy change needs an allocation theory under composition constraints (we contribute a stress test and the η/ν criteria). The teacher is black-box, so deployed distillation uses the trajectory-level hard estimator; the credit weighting did not measurably outperform uniform weighting on the same verified corpus—we claim verified data selection, not the weighting scheme, as the empirical mechanism. (2) Teacher necessity is calibrated against an API frontier model, and paired-counterfactual cost grows linearly with audited events. (3) Conclusions rest on six robosuite families and one physical platform; transfer is selective. (4) The $\pi_{1.5}$ row carries a collection-seat artifact; ablation pools differ from the main-table protocol; the normalization arm is a single run, and selection-ablation arms are single runs per seed (three seeds on lift). (5) The handback design (control returning to the student immediately after each verified patch, §3.2) rests on design argument, and RFT/STaR remains inferred from the gate-off ablation and the self-imitation arm. (6) Stage-entry (ρ) decisions reused the evaluation cohort as validation (validation and test are not fully separated for the reported stage-entry decisions); a disjoint never-used cohort re-evaluation reproduces the base/cold conclusions (Appendix D), and the released protocol separates them prospectively. The reuse is consequential only for thin margins: retained generations cleared $+18$ to $+84$ points and abstentions sat 16–20 points above ρ; the one thin entry (lift, 76% against ρ=80%) is precisely the case the criterion re-measured and rejected (§4.2).

B PROOFS AND STATISTICAL GUARANTEES

Throughout, $\bar{R} := R\,\frac{1-\gamma^T}{1-\gamma}$ bounds any trajectory return (T-step horizon, $|r_t| \le R$), and $S_\gamma := \sum_{t=0}^{T-1}\gamma^t$. Snapshot-restoration and execution-identity discrepancies are booked into $\mathcal{E}_{\mathrm{sys},k}$.

Formal definitions (§3.1).

- *Return.* $J(\pi) = \mathbb{E}_{x\sim\rho,\zeta\sim P^\pi}\big[\sum_{t=0}^{T-1}\gamma^t r_t\big]$.
- *Histories.* At the t-th code decision point, $h_t = (o_0, c_0, o_1, c_1, \ldots, c_{t-1}, o_t)$ with bounded segments $c_t \in \mathcal{C}$ treated as macro-actions; the occupancy is $d^t_{\pi_k}(h_t \mid x) = \Pr_{\pi_k}(H_t{=}h_t \mid x)$.
- *Teacher policy and action values.* The teacher's local policy $\tau_k(\delta \mid x, h_t)$ proposes a bounded correction whose suffix is regenerated by π_k;

$$Q^{\pi_k}_t(h_t,\delta) = \mathbb{E}\Big[\sum_{s\ge t}\gamma^{s-t}r_s \,\Big|\, H_t{=}h_t,\, C_t{=}\delta,\, C_{>t}{\sim}\pi_k\Big], \tag{B.1}$$

  with $V^{\pi_k}_t(h_t) = \mathbb{E}_{c\sim\pi_k}[Q^{\pi_k}_t(h_t,c)]$, $A^{\pi_k}_t = Q - V$, and $\Delta_k(h_t) = \mathbb{E}_{\delta\sim\tau_k}[A^{\pi_k}_t(h_t,\delta)]$.
- *Contract selection.* $\chi^\star_k(\widehat{h}) = \arg\min_{\chi\in\mathfrak{X}(\widehat{h},z)}\kappa(\chi)$ subject to $\widehat{\mathrm{LCB}}_k(\widehat{h};\chi) > \epsilon$ and $\mathrm{Valid}(\chi, \bar{\tau}^\chi_k, s^{\mathrm{snap}}_{\widehat{h}}) = 1$, applied per $\bar{\tau}^\chi_k$ during selection, where Valid requires that the contract-constrained policy support meets execution-safety and program-context constraints and that snapshot-restoration and execution-identity checks agree; with constrained decoding, support constraints are realized implicitly by the decoder.

Two-layer confidence radius. Fix a history $\widehat{h}$ and let teacher candidates $\delta_1,\ldots,\delta_K \overset{\mathrm{iid}}{\sim} \bar{\tau}_k(\cdot \mid x, \widehat{h})$ with paired estimates $\widehat{A}_k(\widehat{h},\delta_j) = \frac{1}{M}\sum_{m=1}^M[G^T_{jm} - G^S_{jm}]$ under shared randomness ξ_m, and $\widehat{\Delta}_k(\widehat{h}) = \frac{1}{K}\sum_j \widehat{A}_k(\widehat{h},\delta_j)$. Since the paired differences lie in $[-\bar{R},\bar{R}]$ with common noise eliminated by pairing, Hoeffding applied to the inner (M-pair) and outer (K-candidate) layers gives, for any $\alpha_1,\alpha_2 \in (0,1)$, with probability at least $1-(\alpha_1+\alpha_2)$,

$$\big|\widehat{\Delta}_k(\widehat{h}) - \Delta_k(h)\big| \;\le\; \underbrace{\bar{R}\sqrt{\tfrac{2\ln(2/\alpha_1)}{K}} + \bar{R}\sqrt{\tfrac{2\ln(2/\alpha_2)}{M}}}_{=\,b_k(\widehat{h};K,M,\alpha)}. \tag{B.2}$$

Simultaneous coverage and selection. Allocate levels $\alpha_{k,h}$ with $\sum_{k,h}\alpha_{k,h} \le \alpha$ across all histories verified in generation k; by the union bound the event $\mathcal{E}_{\mathrm{cov}} := \{\forall(k,h) : \Delta_k(h) \ge \widehat{\Delta}_k(h) - b_k(h;K,M,\alpha_{k,h})\}$ holds with probability at least $1-\alpha$. The admission gate g_k is measurable in $\widehat{\Delta}_k$ (data-dependent selection), but $\mathcal{E}_{\mathrm{cov}}$ holds uniformly over *all* histories, so selection does not invalidate the per-history bound—this is why simultaneous rather than pointwise confidence bounds are used.

Implementation note (analytical vs. operational bound). The Hoeffding radius above is the analytical form used in the theorem statements. The deployed verifier (`anytime_valid_betting_mixture_v1`) uses an anytime-valid e-process over the same paired differences (success and normalized-reward channels in parallel), which enjoys the same coverage guarantee with a tighter radius; the theorem statements are unaffected by this choice. Admission used $\epsilon = 0$ and $\alpha = 0.05$; each audited event starts at $M{=}8$ shared-randomness pairs and escalates ($12/16/24/32/48$) when evidence is insufficient; candidates are drawn under top-k constrained decoding within the selected contract. At the base pairing, one audited event costs up to $(K{+}1)\times M$ suffix rollouts; the full ledger contains 110 audited events.

History–state performance-difference lemma. For any policies π, π',

$$J(\pi') - J(\pi) = \sum_{t=0}^{T-1}\gamma^t\,\mathbb{E}_{h\sim d^t_{\pi'}}\big[\mathbb{E}_{c\sim\pi'(\cdot|x,h)}[Q^\pi_t(h,c)] - V^\pi_t(h)\big]. \tag{B.3}$$

Proof. Induction on T. For $T{=}1$ both sides equal $\mathbb{E}_{h\sim d^0}\mathbb{E}_{c\sim\pi'}[r(h,c)] - V^\pi$. For the step, decompose $Q^\pi_T(h,c) = r(h,c) + \gamma\,\mathbb{E}_{h'}V^\pi_{T-1}(h')$, difference the first-step expectations of π' and π, expand the continuation of π' along $d^t_{\pi'}$, and apply the hypothesis. Only the Markov property of the macro-action history process is used. □

Execution-time identity. Under the mixture of Eq. 3,

$$J(\mu_k) - J(\pi_k) = \sum_t \gamma^t \mathbb{E}_{h\sim d^t_{\mu_k}}\big[g_k(h)\,\Delta_k(h)\big]. \tag{B.4}$$

Proof. Take $\pi' = \mu_k$ in the lemma; per history, the student component $(1-g_k)\pi_k$ cancels against $V^{\pi_k}_t(h)$ and the teacher component contributes $g_k(h)\Delta_k(h)$. □

Execution-time bound (Eq. 4). On $\mathcal{E}_{\mathrm{cov}}$, $g_k(h)\Delta_k(h) \geq g_k(h)\big(\widehat{\Delta}_k(h) - b_k(h)\big)$ per history; on admitted histories $\widehat{\Delta}_k(h) > b_k(h) \geq 0$ so $[\widehat{\Delta}_k(h)]_+ = \widehat{\Delta}_k(h)$, and both terms vanish elsewhere. Summing yields $J(\mu_k) - J(\pi_k) \geq \widehat{\mathcal{G}}_k - \mathcal{E}_{\mathrm{verify},k}$. □

Seat invariance (Proposition 1). The proofs above use only the history distribution induced by the seat policy; replacing π_k by any σ throughout gives $J(\mu^\sigma) - J(\sigma) \geq \widehat{\mathcal{G}}^\sigma - \mathcal{E}_{\mathrm{verify},\sigma}$, with $\mu^\sigma = \sigma$ when $g \equiv 0$ (identical process), and strict inequality when a visited history carries $g(h) > 0$ (its admitted contribution is strictly positive and the verification term is controlled by admission). □

Occupancy divergence chain. For any θ and $t \geq 1$,

$$\mathrm{TV}(d^t_{\mu_k}, d^t_{\pi_\theta}) \leq \sum_{s<t} \mathbb{E}_{h\sim d^s_{\mu_k}}\big[\mathrm{TV}\big(\mu_k(\cdot \mid x,h),\, \pi_\theta(\cdot \mid x,h)\big)\big]. \tag{B.5}$$

Proof. The two history distributions can diverge only after their first divergence point; sum over divergence times and merge same-time contributions by the recursion of the history distributions (standard simulation argument). □

Projection-error bound. Combining the chain with Pinsker ($\mathrm{TV} \leq \sqrt{D_{\mathrm{KL}}/2}$) and Cauchy–Schwarz,

$$|J(\mu_k) - J(\pi_\theta)| \;\leq\; C_{T,\gamma,R}\sqrt{\varepsilon_{\mathrm{proj},k}}, \tag{B.6}$$

with $\varepsilon_{\mathrm{proj},k} = \sum_t \gamma^t \mathbb{E}_{h\sim d^t_{\mu_k}}[D_{\mathrm{KL}}(\mu_k\|\pi_\theta)]$ and $C_{T,\gamma,R} = \bar{R}_{\mathrm{norm}}\sqrt{S_\gamma/2}/(1-\gamma)$ for the appropriately normalized return bound.

Theorem 1. On $\mathcal{E}_{\mathrm{cov}}$, decompose $J(\pi_{k+1}) - J(\pi_k) = [J(\mu_k) - J(\pi_k)] - [J(\mu_k) - J(\pi_{k+1})]$ and bound the first term by the execution-time bound and the second by the projection-error bound at θ of π_{k+1}. □

Token-level decomposition. When teacher and student share a tokenizer, $D_{\mathrm{KL}}(\bar{\tau}_k\|\pi_\theta)$ decomposes into per-token conditional KLs by the chain rule, with the termination symbol included in δ. □

Corollary 1 (cold-start amortization). The cold corpus $\mathcal{D}_0$ consists of *verified-completed* trajectories of P^{μ^τ}, so the full-trajectory empirical risk $\frac{1}{|\mathcal{D}_0|}\sum_\zeta -\log\pi_\theta(\zeta)$ is an unbiased surrogate of the *completion-conditioned* divergence $D_{\mathrm{KL}}(P^{\mu^\tau} \mid C \,\|\, P^{\pi_\theta})$ (C the completion event; trajectory-level chain decomposition), whose population optimum is the projection on the success-conditioned occupancy—not on the unconditioned μ^τ occupancy. This conditioning is the cold-start instance of the same credit weighting: with no per-history gates yet, verification reduces to the completion indicator—the trajectory-level $w \equiv \mathbf{1}\{C\}$ of Eq. 5—and the Theorem-1 decomposition carries the teacher-seat bound of Proposition 1 through with the projection error measured against this weighted target. The gap between the conditioned and unconditioned occupancies is exactly the selection cost that later rounds avoid by conditioning on *local counterfactual* credit instead of trajectory outcomes. □

Corollary 2 (safe abstention). If $\underline{\Delta}_\sigma(h) \le \epsilon$ almost everywhere, then $g_\sigma(h) = 0$, so $\mu^\sigma = \sigma$ per history (no execution-time change) and the training weight $g_\sigma(h)w_\sigma(h) \equiv 0$ (no teacher term): neither execution nor training introduces unverified teacher credit. □

Remark (rewind semantics, full). The deployed pipeline runs two passes: the student's live attempt realizes a failure and localizes candidate boundaries; verification then rewinds to the stored snapshot $s^{\mathrm{snap}}_{\widehat{h}}$. We analyze the *post-rewind* process: $J(\mu_k)$ measures the return of the repaired continuation from $s^{\mathrm{snap}}_{\widehat{h}}$ onward, and both arms of every paired comparison (teacher correction vs. student resample) start from the *same* snapshot under shared subsequent randomness. The first pass is therefore a fixed data-collection cost common to the comparison and cancels in Eq. 4; equivalently, μ_k is a one-pass policy over histories whose state includes the snapshot stack. Formally, the augmented process has state $(s, \mathcal{Z})$ with $\mathcal{Z}$ the snapshot–evidence stack: the diagnostic evidence z observed after the boundary is part of the state that *selects* the intervention point, so whether to intervene at $\widehat{h}$ depends on information observed after $\widehat{h}$ in the original rollout. $J(\mu_k)$ is the expected return of the post-rewind continuation under this two-pass process; it is not the return of a causal online policy in the original environment, and the first pass cancels only because both arms of each paired comparison share it as a fixed cost. Theorem 1 bounds the augmented-process return; a snapshot-free deployment that diagnoses from the current history alone—as our hardware deployment does, running the standalone student with no teacher calls—is outside this guarantee.

Ideal supervision allocation. Given a budget B of teacher calls and verification rollouts, the ideal allocation solves, over histories,

$$\max_{q_k} \sum_t \gamma^t \, \mathbb{E}_{h\sim d^t_{\pi_k}} \big[q_k(h)\, \Delta_k(h) \big] \quad \text{s.t.} \quad \sum_t \gamma^t \, \mathbb{E}_{h\sim d^t_{\pi_k}} \big[q_k(h)\, \kappa(h) \big] \le B, \;\; q_k(h) \in [0,1], \tag{B.7}$$

a fractional-knapsack linear program requiring $\Delta_k(h)$ and the occupancy, neither directly observable. RE-0's diagnostic localization (shrinking the candidate history set) and LCB admission (accepting only verifiable positive credit) form its practical proxy: in the empirical-LCB sense, the admitted set is exactly the histories with estimated positive credit.

C Experimental details and reproducibility

Training and evaluation configuration. All students are Qwen3.8-27B with LoRA (r=16, α=32, all-linear), trained with the trajectory-level hard estimator for 120 steps (batch 2, LR 10^{-5} constant, seed 1, bf16); refinement warm-starts from the previous generation at LR 5×10^{-6}. Evaluation uses 50 held-out seeds per task (temperature 0.2, top-p 0.95, non-thinking mode) reset seeds from the 210000001 series and generation seeds from the 220000001 series (stride 10007), disjoint from collection. Collection verifies each contract with K teacher candidates against M shared-randomness paired rollouts; admission requires the two-layer LCB of Eq. 2 to exceed ϵ under the α-allocated simultaneous budget.

Verification cost. Across the six tasks the full ledger contains 110 audited events, of which 19 were admitted. Each event verifies its selected candidate from the boundary snapshot with M shared-randomness pairs (escalating $8/12/16/24/32/48$ when evidence is insufficient; observed distribution $27/50/3/8/1/21$ events, median M=12), each pair executing one student resample and one teacher continuation: the selected candidates alone consumed $4{,}192$ suffix rollouts in total. Rejected candidates and the live first pass add further environment rollouts, so verified rows are information-dense but not rollout-free.

D Main-result accounting and robustness

Table 1 notes. (a) Base row uses the standard 50-seed evaluation cohort, disjoint from collection. (b) The refine cell for restack comes from an intermediate teacher-seat recollection; a collection-seat artifact affecting this cell is disclosed in the Limitations. (c) Spill saturates at stage one; its counterfactual admission set is empty, so the criterion abstains—the runtime instance of Corollary 2. (d)

Lift refinement consumed a single admitted event (w=0.069) plus replay: 76%→58%, with collateral collapse on the branch (spill $28 \to 0$)—headroom without credit mass is harmful (Remark 2). (e) Cubestack is saturated at base; the system exempts it from training. The transfer table's own-task cells for π_1/π_2' (restack) are 31/40 (Table 1). (f) Composition arms (restack, 50 seeds): plain SFT on 8 success-filtered teacher-seat demonstrations scores 3/50; SFT on outcome-selected patches (3 rows, unweighted) scores 46/50. The nine patches rejected by the outcome rule all have intervention success 0.0—zero divergence from the LCB gate on this corpus; both arms are single runs. (g) Row counts and admission counts are different objects: cold rows count verified-completed *teacher-seat* trajectories (the Corollary 1 corpus), while Figure 3b counts admission events in *student-seat* collection rounds.

Per-round sign accounting. Theorem 1 bounds the standalone gain by the verified intervention gain net of verification and projection terms; the first two are measured at admission, while the projection term is guarded operationally by the outer acceptance criterion (Remark 1). Table 3 reports, for every round in which the criterion made a retain-or-halt decision, the admitted credit of the round against the realized change in standalone performance on the 50-seed protocol. The signs agree in all four rounds, including the round the criterion rejected. We read this as sign-level consistency evidence over four rounds, not an estimate of $\varepsilon_{\mathrm{proj},k}$. Re-executing the rejected lift round with training seed 2 reproduces the regression (76%→46%), so the halt reflects the round's credit rather than a seed artifact.

Round	Admitted credit	$\widehat{J}$ change	Decision	Retained
restack $\pi_0 \to \pi_1$ (cold)	6/6 events, $\widehat{\Delta} \approx +1.0$	$+54$ pt ($8 \to 62\%$)	$J < \rho$: continue	π_1
restack $\pi_1 \to \pi_2'$ (refine)	4/15, $\widehat{\Delta} \in [0.19, 0.88]$	$+18$ pt ($62 \to 80\%$)	$J \geq \rho$: halt	π_2'
lift $\pi_1 \to \pi_2$ (refine)	1 event, w=0.069	-18 pt ($76 \to 58\%$)	regression: halt	π_1
spill $\pi_1 \to$ (refine)	0 admitted	— (no projection)	abstain (Cor. 2)	π_1

Table 3: Per-round sign accounting: admitted credit against realized standalone change. The retained generation is the measured-better one in every round.

Task	Base	RE-OPD-cold	RE-OPD-refine
restack	[3.2, 18.8]	[48.2, 74.1]	[67.0, 88.8]
spill	[54.2, 79.2]	[92.9, 100.0]	—
nut	[1.1, 13.5]	[76.2, 94.4]	—
lift	[1.1, 13.5]	[62.6, 85.7]	[44.2, 70.6]
door	[86.5, 98.9]	[92.9, 100.0]	—
cubestack	[92.9, 100.0]	[92.9, 100.0]	—

Table 4: Wilson 95% CIs (%) for Table 1.

Wilson confidence intervals.

Trainer-seed replication. We retrained the restack and spill cold-start arms with two additional trainer seeds on the identical corpora and recipe. Restack: seed 1 = $31/50$, seed 2 = $44/50$, seed 3 = $36/50$ (mean 37.0)—every replicate far above the base 4/50. Spill: seed 1 = $50/50$, seeds 2 and 3 = $33/50$ each (mean 38.7)—two of three replicates sit at base level (34/50), so the spill cold-start gain is seed-sensitive. With the same protocol we additionally replicated the remaining cold arms: lift reaches $38/33/35$ (mean 35.3), nut $44/40/43$ (mean 42.3), and door $50/50/50$—every replicate sits far above its base, so the large cold-start gains on the thin-corpus tasks are robust to trainer seed, and spill remains the sole seed-sensitive task.

Never-used cohort re-evaluation. Base and cold checkpoints were additionally re-evaluated on a disjoint, never-used 50-seed cohort (reset offset 230000001, generation offset 240000001): base $3/3/5/47/50$ (restack, nut, lift, door, cubestack) and cold $28/45/42/50/49$ (restack, nut, lift, door, spill) match the reported cells within binomial noise, confirming that the reported numbers are not artifacts of the evaluation-seed protocol.

E MECHANISM AND ABLATION DETAILS

Table 2 holds each task's collection journal fixed and varies only the admission rule. Provenance: the lift/spill/nut journals are teacher-seat (self-play) collections; all arms use identical row construction (uniform weight, full-trajectory SFT). The four lift events admitted by the sign rule ($\widehat{\Delta} > 0$) but rejected by the LCB have $\widehat{\Delta} \in \{+0.21, +0.06, +0.04, +0.00004\}$ at $M \in \{48, 48, 48, 12\}$: positive in mean, statistically indistinguishable from zero. Table 5 completes the grid with direction cells and the restack self-imitation arm.

Task / rule	Rows	Seed 1	Seed 2	Note
lift: none	25	6	2	
lift: sign	7	5	4	
lift: LCB	3	**11**	**5**	
lift: outcome	27	5	9	
spill: none	6	10	—	mean reward 0.63
spill: sign	2	0	—	collapse; reward 0.00
spill, nut: LCB	0	—	—	abstain
nut: none	4	0	—	
nut: sign	3	2	—	
restack: self-imitation	11	2	—	RFT analogue

Table 5: Full selection-ablation grid (successes/50; one run per seed).

The spill sign arm collapses to 0/50 with exactly zero reward: all fifty rollouts execute the same three-block program with no wipe coverage—two near-duplicate rows reduce the policy to template repetition. The LCB admits nothing on spill or nut and abstains. True base-seat RFT is infeasible in our protocol; the nearest analogue—self-imitation on the 11 success programs of the π_1-seat policy—reaches 2/50 under identical row construction.

Teacher-effect ladder. Table 6 details the admission-ledger grading: student self-play yields no verifiable credit; narrow local task experts (win rates 62–100% on their own tasks) fail deterministically at the contract-format layer with a $\sim$10% rescue ceiling; the strong API teacher earns 4 admissions on 13 events with rescue rates 19–47%.

Table 6: **The admission ledger as a teacher-effect instrument.**

Teacher	Events	Admitted	Rescue rate	Failure mode
Student self-play	33^{+}	**0**	$\sim$0	no credit at $p \approx 0$
Local task experts	32	**0**	$\sim$10% ceiling	contract-format failures
API frontier	13	**4**	19–47%	—

$^{+}$restack 11, nut 6, spill 5.

F TRANSFER, POOLING, AND DEPLOYMENT

Table 7: **Cross-task transfer** (successes / 50, @120).

Policy \ Task	nut	spill	lift	handover
Base	2	34	2	—
π_1 (restack student)	22	24	6	0
π_2' (restack refined)	16	**45**	3	1
Lift expert	0	28	**38**	0

Own-task (restack): 31/40.

Real-robot platform. The deployment platform is a dual-arm AgileX PiPER rig with SAM3-based open-vocabulary perception (Figure 5); the execution stack exposes the simulation API surface unchanged, so the deployed policy code requires no hardware-specific edits.

Table 8: **Pooled joint training fails** (successes / 50). Two separable failures: task interference and quota starvation.

Task	Expert	Equal (42)	Proportional (25)	Full-pool (172)	Credit-equal
restack	**31**	3	2	2	1
spill	**50**	31	0	0	32
nut	**44**	9	0	0	21
lift	**38**	20	25	5	15
cubestack	50	50	0	48	48
door	50	50	0	49	49

Compute infrastructure. All simulation training, collection, and evaluation ran on an eight-GPU NVIDIA H200 server (143 GB per GPU), one GPU per run: the 27B student is served with vLLM (LoRA adapters, runtime hot-loading), training uses FSDP with gradient checkpointing, and verification rollouts run against the same engine; teacher and diagnosis calls go to an external API model.

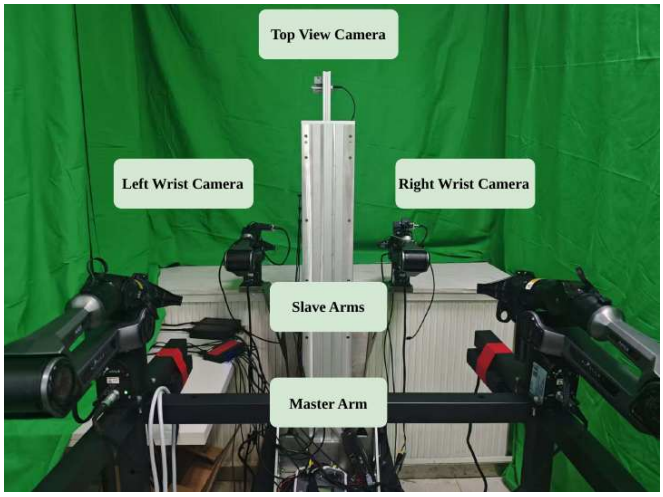

Figure 5: **Real-robot platform.** The dual-arm AgileX PiPER rig used for the zero-shot deployment experiments.

G BASELINE DETAILS

GRPO configuration and fairness. The GRPO control was tuned until it trained at all in this stack: chat-template thinking mode disabled (otherwise every response clips at the token budget and all rewards are zero), max response length 3072, thread-safety fixes for the robosuite reward and IK-server liveness monitoring, and memory-feasible two-card colocate settings for the 27B policy. KL regularization (reference model) was disabled to fit GPU memory; entropy collapse ($0.30 \rightarrow 0.002$) persisted across all of these fixes. We cannot rule out that a larger-memory configuration with KL regularization behaves differently; the comparison should be read as against a resource-matched GRPO, not a maximally tuned one. The GRPO control evaluates on the same 50-seed protocol: restack 1/50, spill 0/50, lift 0/50, cubestack 50/50 (base 4/34/2/50); the nut evaluation could not be completed due to a serving-stack incompatibility and is omitted.