AUDITING LEARNED ROBOT REWARDS

Anonymous authors

ABSTRACT

Learned visual rewards can score near-goal failures highly, allowing PPO to increase proxy reward while task success falls. Fixed-P8 runs show a 9.9 pp ManiSkill peak-to-final success decline and a matched-proxy residual that turns negative. Label-Selective Audit and Refresh (LSAR) uses random gold audits of top-scored trajectories and refreshes the reward when their precision falls below a reference level. At a 1,000-label cap, LSAR ends at 72.3% on ManiSkill and 40.8% on LIBERO-Long, versus 66.9% and 33.4% for periodic top-score labeling, using a mean of 765 and 637 method-requested gold labels, respectively; shared setup and diagnostic labels are excluded. Shadow precision, held-out audits, matched-cost controls, and component analyses characterize the refresh mechanism and its label cost.

1 INTRODUCTION

Consider a robot that must place an object on a plate. It holds the object a few centimeters above the plate and never opens the gripper. A learned progress reward that reads the last few camera frames can score this state almost as highly as a completed placement, because the images differ only in a small gap and a finger configuration. Optimized against that reward, the policy can learn to hover while the proxy keeps rising and completed placements stall or decline. Reward misspecification and overoptimization are well documented in reinforcement learning and language-model alignment (Pan et al., 2022; Skalse et al., 2022; Gao et al., 2023), but for robot policies adapted with learned visual rewards the corresponding evidence is thin: recent reward models report offline accuracy or value-order correlation (Lee et al., 2026; Liang et al., 2026; Zhai et al., 2025; Zhang et al., 2025), rarely how often their high scores are failures during optimization.

Two decisions follow when success and proxy diverge during adaptation. The first is diagnostic: how success under the adapted policy compares with success attainable by a specified reweighting of labeled initial-policy trajectories at the same proxy mean. The comparison identifies a residual to that reference; labels on current behavior are needed to assess its high-score failures directly. The second concerns cost. Gold labels are the direct evidence of the gap, and we impose a cap on privileged simulator-predicate queries beyond shared setup. A fixed schedule spends that budget whether or not the reward has degraded, while label-free mitigations such as a KL penalty or a conservative ensemble alter optimization without measuring success (Ziegler et al., 2019; Coste et al., 2024; Eisenstein et al., 2024).

We address the first decision with a selection reference (Figure 1). Exponential tilting of a gold-labeled pool of initial-policy episodes, scored by the same reward, traces the success reachable by reweighting them toward higher scores against the proxy mean. A checkpoint's matched-proxy residual is its success minus the reference success at its own proxy mean, reported only where the finite pool retains an adequate effective sample size; a negative value indicates that the checkpoint succeeds less often than a reweighting of initial behavior with the same average score.

We address the second decision with Label-Selective Audit and Refresh (LSAR). At fixed training boundaries, LSAR queries gold labels in stages on a random sample from the top-scored fifth of recent training episodes, comparing a Beta posterior on their precision with the precision the initial reward achieved on the labeled pool. When the posterior indicates a drop, LSAR acquires additional labels ranked by score, ensemble disagreement, and novelty, and refreshes the reward on all labels paid for so far; otherwise the reward is left unchanged.

The evaluation adapts OpenVLA-OFT (Kim et al., 2025) with PPO in RLinf (Zang et al., 2025) on a ManiSkill placement task and the ten LIBERO-Long tasks (Tao et al., 2024; Liu et al., 2023), with

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.

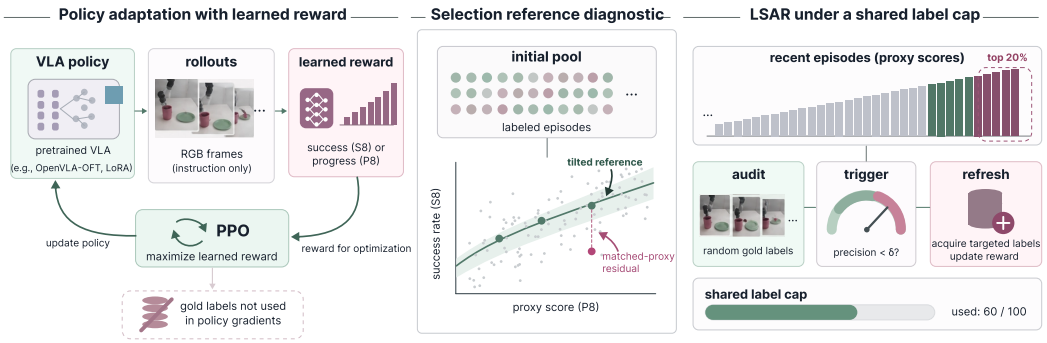

Figure 1: Overview. A VLA policy is optimized with PPO against a learned success (S8) or progress (P8) reward; gold labels do not enter policy gradients in the learned-reward arms. Offline, a gold-labeled pool of initial-policy episodes defines a selection reference against which fixed-reward checkpoints are compared at the same proxy mean. Online, LSAR audits randomly chosen top-scored episodes and refreshes the reward only when their gold precision falls below the initial level; every audit and acquisition label is charged to one cap.

a terminal success classifier or a stage-progress regressor on the same backbone (Bai et al., 2025). The refresh comparison fixes the label cap, reward initialization, update objective, and policy budget, and includes front-loaded labeling, label-free regularization, and periodic schedules over uniform, highest-scored, or highest-disagreement episodes.

Our contributions are as follows.

- A matched-proxy selection diagnostic with an exact finite-pool estimator, an effective-sample-size validity rule, and a split of success changes into reference and residual terms (Section 3.2).
- LSAR, a label-budgeted refresh rule whose sequential random audit of top-score precision decides when targeted labels and a reward update are purchased (Section 3.3). On ManiSkill, LSAR reaches 72.3% final success while spending a mean of 765 of a 1,000-label cap.
- A shared-cap comparison of audit-triggered and scheduled refresh (Section 4). LSAR exceeds the top-score periodic baseline by 5.5 pp on ManiSkill. Shadow precision, LSAR audit ledgers, and held-out post-refresh precision provide complementary diagnostics.

2 Related Work

Reward Overoptimization. Gao et al. (2023) measure how a gold reward first rises and then falls as best-of-n sampling or RL optimizes a proxy reward model. Formal accounts characterize when proxy optimization must hurt the true objective (Skalse et al., 2022; Karwowski et al., 2024), and the Goodhart taxonomy distinguishes regressional and extremal effects (Manheim & Garrabrant, 2018). Our matched-proxy residual describes departure from a reweighting of the initial labeled trajectories; it does not identify either mechanism. Reward-model ensembles reduce some overoptimization but share errors across members (Coste et al., 2024; Eisenstein et al., 2024), so we use disagreement only to rank candidates for labeling and let a random gold audit decide whether to refresh.

Selection with Imperfect Verifiers. Best-of-n selection and KL-regularized optimization share an exponential-tilt structure (Beirami et al., 2025; Yang et al., 2024; Gui et al., 2024). With an imperfect verifier, best-of-n success can saturate or fall as n grows, depending on the verifier's ROC curve and the base policy's coverage (Huang et al., 2025; Dorner et al., 2026; Stroebl et al., 2026); for robot policies, selection by an action verifier saturates while selection by a learned value function falls at large candidate counts (Kwok et al., 2025; Nakamoto et al., 2025). We use the tilt of a finite labeled

pool as a reference distribution within a validated proxy range, not as a claim that PPO realizes the tilt.

Reward Refresh and Active Querying. Iterated RLHF retrains the reward model on each round's policy samples (Ouyang et al., 2022; Wolf et al., 2025), and preference-based robot learning queries labels on a fixed schedule, often over ensemble-disagreement or recent on-policy pairs (Lee et al., 2021b;a; Hu et al., 2024). VICE-RAQ trains a success classifier as the reward and actively queries a user about states the classifier selects during robot RL (Singh et al., 2019), InfoRM detects overoptimization from latent outliers without gold labels (Miao et al., 2024), and drift adaptation contrasts scheduled with error-triggered retraining (Gama et al., 2014). LSAR triggers on gold precision in the top-score stratum, measured by random sampling and charged to the same budget as acquisition; the periodic top-score schedule provides a relevant comparison with a different update schedule, acquisition rule, and realized label spend.

Reinforcement Learning for VLA Policies. PPO and GRPO fine-tuning of VLA policies with simulator success rewards improves success on ManiSkill and LIBERO, in distribution and out of it (Liu et al., 2025; Zang et al., 2025; Li et al., 2026; Tan et al., 2025; Lu et al., 2025). Where predicates are unavailable, vision-language reward progress models trained on robot et al., 2024; Zhang et al., 2025; Zhai et al., 2025; Lee et al., 2026; Liang et al., 2026), and their quality is evaluated mainly before or alongside training. Our fixed-reward runs characterize failures during optimization, while LSAR evaluates an audit-triggered correction and its label cost.

3 Selection Reference and Label-Budgeted Refresh

3.1 Setting

Let x denote an initial state and instruction and τ a complete episode. The gold label $g(\tau) \in \{0, 1\}$ records task completion under a simulator predicate that the policy and reward never observe, and the proxy $s(\tau) \in [0, 1]$ is a fixed normalization of the learned reward's discounted return. S8 predicts terminal success from the last observations; P8 predicts cumulative stage progress at regular intervals, so a policy can receive most of its return before the final predicate holds (Appendix B.4). The policy π_θ is optimized with PPO on s only, and a method may issue at most B unit-cost gold queries beyond the data that fits the initial reward.

3.2 Selection Reference

The reference asks how much success is available by preferring higher-scored episodes of the unchanged initial policy π_{ref}. For each state, exponential tilting reweights initial-policy episodes by their proxy,

$$q_\lambda(\tau \mid x) = \frac{\pi_{\text{ref}}(\tau \mid x)\, e^{\lambda s(\tau)}}{Z_x(\lambda)}, \qquad \lambda \geq 0, \tag{1}$$

and $S(\lambda) = \mathbb{E}_x \mathbb{E}_{q_\lambda}[g]$ and $R(\lambda) = \mathbb{E}_x \mathbb{E}_{q_\lambda}[s]$ trace success against proxy mean with the state distribution fixed. For $\lambda > 0$ the tilt maximizes expected proxy minus λ^{-1} times the KL divergence from π_{ref} over distributions absolutely continuous with respect to it (Appendix A.2); this is a distributional reference, not a policy that PPO is guaranteed to reach.

Whether selection can hurt depends on how scores separate successes from failures. At a state with base success $p_x \in (0, 1)$, let $\ell_x(s)$ be the likelihood ratio of the score under success versus failure.

The tilted success odds are

$$\frac{S_x(\lambda)}{1 - S_x(\lambda)} = \frac{p_x}{1 - p_x}\, \mathbb{E}_{F,\lambda}[\ell_x(s)], \tag{2}$$

where the expectation is over failed episodes tilted by their score. If ℓ_x is nondecreasing, success cannot decrease with λ. Near-goal failures scored as highly as completions can flatten or reverse score separation near the top; a flattening best-of-n curve is compatible with this pattern (Figure 4a).

We estimate the reference on an analysis pool of $M = 100$ states with $m = 128$ initial-policy episodes each, normalizing weights within a state and averaging states equally. The effective sample size of the pooled weights falls from $Mm = 12{,}800$ at $\lambda = 0$ toward M as every state concentrates on one episode (Kong et al., 1994); we evaluate 241 tilts and keep those with ESS ≥ 500 (Appendix A.5).

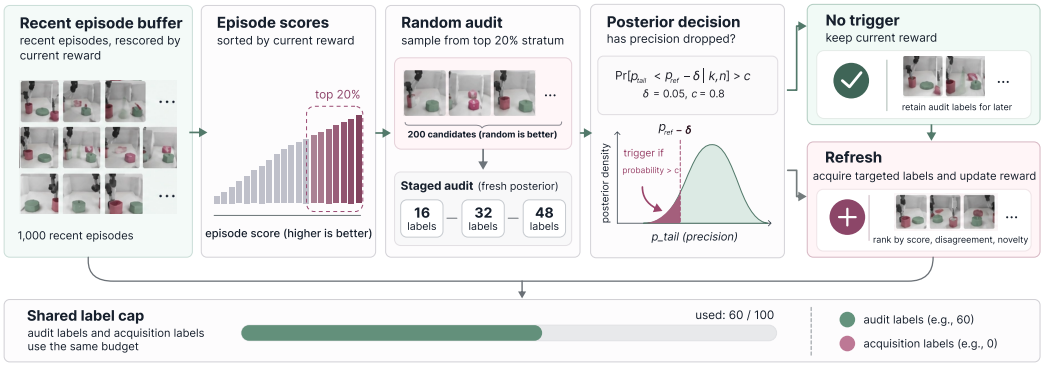

Figure 2: LSAR decision rule at one audit boundary. A random sequential audit of the top-scored stratum estimates gold precision; only a posterior drop below $p_{\text{ref}} - \delta$ purchases targeted labels and a reward update. Labels from audits that do not trigger enter the next refresh.

Let $\widetilde{S}(r)$ interpolate the valid pairs $(\widehat{R}(\lambda_j), \widehat{S}(\lambda_j))$. A checkpoint with success G_θ and proxy mean $\bar{R}_\theta$ under the same reward version has matched-proxy residual

$$\Delta_{\text{sel}}(\theta) = G_\theta - \widetilde{S}(\bar{R}_\theta), \tag{3}$$

reported only when $\bar{R}_\theta$ lies between the untilted pool mean and the largest valid tilted mean; points outside that range are unmatched, not zero. For two matched checkpoints a and b,

$$G_a - G_b = \big(\widetilde{S}(\bar{R}_a) - \widetilde{S}(\bar{R}_b)\big) + \big(\Delta_{\text{sel}}(a) - \Delta_{\text{sel}}(b)\big). \tag{4}$$

The first term is the change a pure reweighting of initial behavior would produce between the two proxy levels; the second is what the reference does not account for. A falling residual after the success peak means checkpoint success has fallen relative to this reweighting reference at the same proxy mean; it does not identify a new failure mode or establish that the policy has left the initial support (Appendix A.6).

3.3 Label-Selective Audit and Refresh

LSAR directly monitors gold success precision among the highest-scored recent episodes. Before training, the initial reward's top-20% stratum on the analysis pool defines a fixed target p_{ref}. At each audit boundary, LSAR rescores the 1,000 most recent complete training episodes with the current reward, forms the stratum of its 200 highest scores, and queries gold labels on randomly ordered members of that stratum in stages of 16, up to 48 labels. After k successes among n audited episodes, a uniform prior gives $p_{\text{tail}} \mid k, n \sim \text{Beta}(1 + k, 1 + n - k)$, and the audit triggers when

$$\Pr[p_{\text{tail}} < p_{\text{ref}} - \delta \mid k, n] > c, \qquad \delta = 0.05,\ c = 0.8. \tag{5}$$

Auditing stops early without a trigger once this probability falls below 0.5; a nontriggering boundary can therefore use 16, 32, or 48 labels. Each audit starts a fresh posterior; because audits repeat, the buffer mixes policy versions, and p_{ref} is estimated, c is not a calibrated false-alarm rate.

When the trigger fires, LSAR acquires up to 128 further unlabeled episodes from the buffer, ranked by the product of their within-buffer ranks in proxy score, disagreement among five reward heads, and embedding distance to the reward's training episodes; these labels are atypical by construction, so only the random audit enters the trigger. The reward is then updated for 300 optimizer steps on its original fitting data and all gold labels paid for so far:

$$\mathcal{L}_{\text{refresh}} = \mathbb{E}_{\mathcal{D}_0}\, \mathcal{L}_{\text{progress}} + \mathbb{E}_{\mathcal{D}_A}\, \mathcal{L}_{\text{success}}. \tag{6}$$

Binary labels supervise an auxiliary success head that shares the reward's representation; the progress heads keep their targets and still produce the policy reward (Appendix B.8). Every refresh uses

Table 1: Compared methods. Progress-reward methods share the policy budget and initial reward; reward-refresh methods use the same update objective and a cap of $B = 1{,}000$ gold labels. Boundaries are fractions of training.

Method	Labels and Updates	Gold Use
Gold reward	PPO on the simulator predicate	every episode
Fixed S8, fixed P8	learned reward never updated	none
Front-loaded	B initial-policy labels, one update before PPO	B
Periodic (uniform, top score, or disagreement)	$B/8$ uniform, highest-scored, or highest-disagreement buffer episodes at 0.16, 0.26, …, 0.86	B
KL-regularized	fixed P8, per-token KL penalty to the initial policy	none
Conservative ensemble	fixed P8, head mean minus one head standard deviation	none
LSAR	audits at 0.16, 0.24, …, 0.96; acquisition and update on trigger	at most B

Equation 6 with the same steps per update. The arms still differ in their acquisition rules, update schedules, and total labels spent. Labels are charged at request against B, and the final request is truncated to the remaining balance.

4 Experiments

4.1 Experimental Setup

Tasks and Policies. ManiSkill uses the `PutOnPlateInScene25Main-v3` placement task with the RLinf OpenVLA-OFT checkpoint trained for it (Zang et al., 2025; Tao et al., 2024), and LIBERO-Long all ten `libero_10` tasks with the RLinf LIBERO-130 checkpoint (Liu et al., 2023). The policy reads one RGB view and the instruction, samples discretized action chunks, and adapts LoRA weights and a value head with PPO (Schulman et al., 2017). Budgets are 1M environment steps on ManiSkill and 10M on LIBERO-Long, about 50 and 20 PPO epochs of the RLinf recipes, and gold success requires the task predicate to hold, with the object released, for the final ten control steps (Appendix B.2).

Rewards and Pools. S8 and P8 share a Qwen3-VL-8B backbone with LoRA and five bootstrapped linear heads (Bai et al., 2025), each trained once per task group on 8,000 initial-policy clips, selected using 2,000 validation clips, and shared by every seed and method. The analysis pool holds 128 initial-policy episodes on each of 100 states disjoint from reward-training and evaluation states; it supplies the selection reference, p_{ref}, and $R^{\star}$, and never enters a gradient.

Evaluation. Each configuration has five independent training seeds and 12 evaluation checkpoints, each on 256 fixed ManiSkill states or on the 50 benchmark initial states of every LIBERO-Long task. Baselines share state identities. Their 95% intervals use 2,000 bootstrap replicates that resample states jointly and seeds separately per method (Efron & Tibshirani, 1994; Agarwal et al., 2021). LSAR and component comparisons use the same task groups and evaluation budgets.

Endpoints. The primary endpoints are final success differences between LSAR and fixed P8, uniform periodic, and top-score periodic within each task group, in percentage points. Secondary endpoints are peak-to-final change, labels spent, and success per label spent, from a ManiSkill budget sweep with caps of 250-2,000. A fixed reward is classified as degrading on a group when its mean final-checkpoint success falls at least 5 pp below its highest checkpoint mean. The threshold classifies degradation from the observed success curves. That peak is located with gold evaluation, so a peak-to-final change summarizes a run rather than scoring a method, and refresh is read separately in degrading and non-degrading groups.

4.2 Fixed-Reward Optimization

Figure 3 compares three optimization targets from the same initial policy. On ManiSkill, gold-reward PPO raises success from 24.7% to 77.6% within 1M steps, fixed P8 reaches 51.3% after 0.36M steps and ends 9.9 pp lower at 41.4%, and fixed S8 reaches 47.7% and ends at 41.8%; both fixed rewards meet the degradation criterion here. On LIBERO-Long neither does: gold-reward PPO is still improving at the end of the budget (48.6%), fixed P8 declines 4.0 pp from its highest checkpoint,

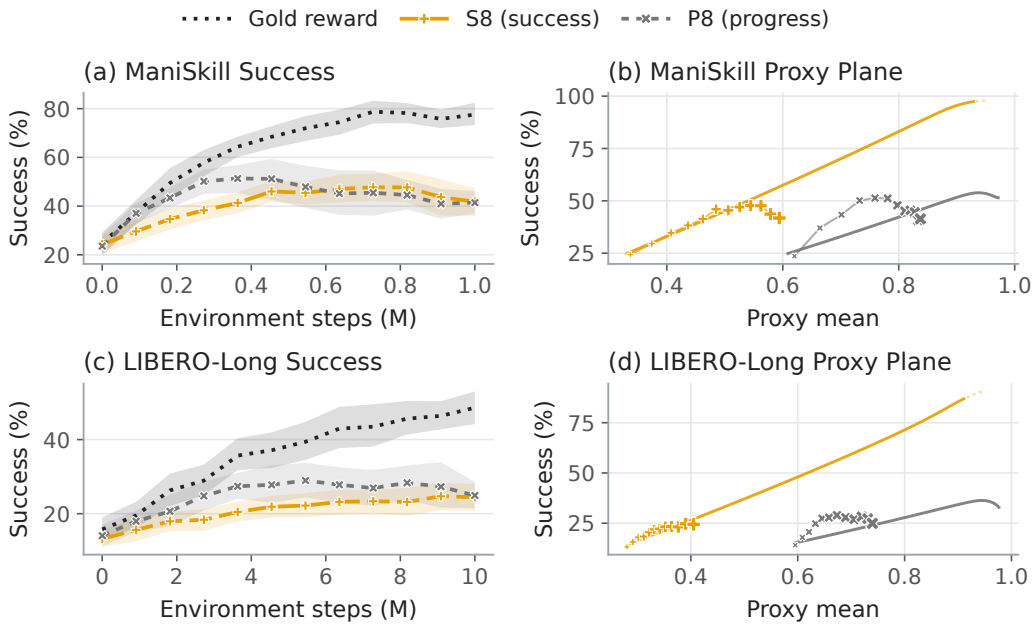

Figure 3: Fixed learned rewards. (a,c) Success over training for gold-reward PPO, fixed S8, and fixed P8; lines average five seeds, bands are 95% bootstrap intervals over seeds and evaluation states. (b,d) The fixed S8 and P8 checkpoints in the gold-versus-proxy plane, with each reward's exponential-tilt reference solid inside its ESS-valid range and dotted beyond it; marker size increases with training step. The S8 and P8 proxy scales are not comparable.

and fixed S8 declines 0.3 pp. Thus LIBERO-Long offers less observed late degradation for refresh to address, without identifying why the task groups differ.

The proxy plane separates a rising score from a rising success rate. Under ManiSkill P8 the checkpoints track the tilt reference while the proxy mean rises to 0.76, then move right of it and below: the final checkpoint has the run's highest proxy mean, 0.84, while success has fallen to 41.4%. LIBERO-Long P8 stays above its reference, without ManiSkill's late drop below it.

4.3 Selection Diagnostics

Figure 4 compares fixed-pool selection with checkpoint success and separately tracks high-score precision during optimization. Selecting the best of n initial-policy episodes by S8 raises ManiSkill success from 24.8% to 97.0% at $n = 64$, whereas selecting by P8 reaches 52.5% at $n = 8$ and stays between 51.0% and 58.0% through $n = 128$; LIBERO-Long gives 90.0% against 36.0% at $n = 128$. The P8 best-of-n curve is consistent with weaker separation of completions from high-scored failures in the initial pool. The finite-pool curve alone does not identify the score likelihood ratio in Equation 2.

The residual measures how the trained policy differs from the specified pool reweighting at a matched proxy mean. Along the ManiSkill fixed-P8 run it is positive through mid-training, peaking at $+14.2$ pp at 0.27M steps, then falls to -4.4 pp at the final checkpoint (95% interval $[-10.1, +1.3]$). Applying Equation 4 between the peak and the final checkpoint splits the -9.9 pp change into a reference term of $+7.4$ pp and a residual term of -17.3 pp: at the proxy mean the final checkpoint reaches, reweighting initial behavior would have gained success and the run instead lost it. The terms have opposite signs, so neither is a share of the decline. Fixed S8 ends with the more negative residual, -14.9 pp $[-22.3, -7.4]$, while losing less success, because its proxy rises into a range where the reference promises much more. On LIBERO-Long the P8 residual stays positive at matched checkpoints and the S8 residual ends at -2.6 pp $[-8.3, +2.4]$, consistent with the absence of a decline there.

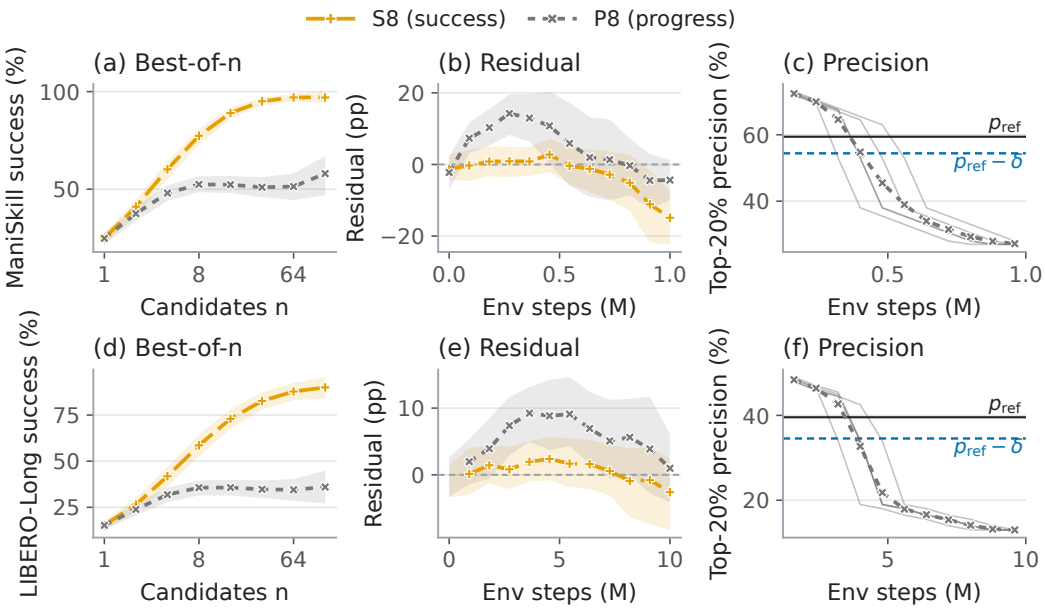

Figure 4: Selection diagnostics for ManiSkill (a-c) and LIBERO-Long (d-f). (a,d) Best-of-n success on the analysis pool with uniform tie breaking; bands resample analysis states. (b,e) Matched-proxy residual of fixed-reward checkpoints, valid matches only; bands resample seeds and both state sets. (c,f) Gold precision among the top-scored 200 of 1,000 recent fixed-P8 episodes. Thin lines are seeds; horizontal lines mark p_{ref} and $p_{\text{ref}} - \delta$.

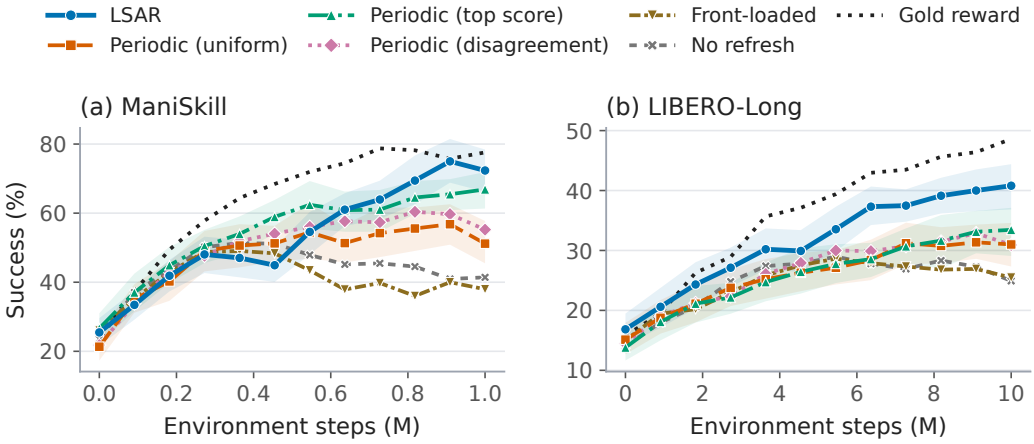

Figure 5: Progress-reward learning curves under a cap of 1,000 gold labels. Curves show five-seed means. Shading for LSAR, uniform periodic, and top-score periodic shows pointwise 95% percentile-bootstrap intervals from 2,000 replicates, resampling evaluation states jointly across methods and training seeds independently by method; each sampled seed carries its full checkpoint trajectory. Gold-reward PPO and fixed P8 are references; boundaries appear in Table 1.

In panels (c) and (f), ManiSkill shadow precision first falls below $p_{\text{ref}} - \delta$ at fractions 0.32-0.56, just after each fixed-P8 success peak. LIBERO-Long shadow precision also crosses the cutoff, although its fixed-P8 success does not meet the 5 pp decline rule. Shadow strata use counts out of 200; the LSAR posterior trigger is evaluated separately on LSAR audit samples.

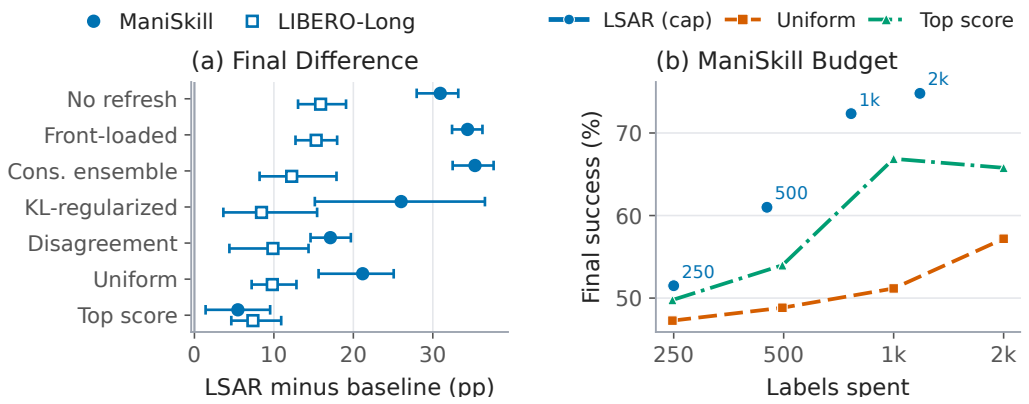

Figure 6: Label efficiency. (a) Final-checkpoint success of LSAR minus each baseline at a cap of 1,000 labels, with seed-resampling ranges. (b) ManiSkill final success against the mean labels actually spent, for caps of 250, 500, 1,000, and 2,000; LSAR points carry their cap because the trigger can leave part of it unspent.

4.4 Refresh under a Label Budget

The methods in Figure 5 share the policy budget and reward initialization. At the 1,000-label cap, LSAR ends at 72.3% on ManiSkill, against fixed P8 at 41.4%, uniform periodic at 51.2%, and top-score periodic at 66.9%. The LSAR ledger spends 765 labels over 4.6 refreshes on average. Each first trigger falls one audit boundary after the corresponding fixed-P8 shadow cutoff, at fractions 0.40-0.64.

On LIBERO-Long, LSAR ends at 40.8% versus fixed P8 at 24.9%, uniform periodic at 31.0%, and top-score periodic at 33.4%, with 637 labels and 3.6 refreshes on average. Fixed P8 does not meet the prespecified 5 pp degradation criterion in this group. The gain therefore cannot be attributed only to recovery from a late success decline; improved reward ranking and acquisition may also contribute.

Figure 6b shows LSAR budget outcomes of 51.5, 61.0, 72.3, and 74.8% at caps of 250, 500, 1,000, and 2,000, spending 250, 450, 765, and 1,180 labels. Appendix D.6 compares top-score and rank-product controls with per-seed label spending and refresh counts matched to LSAR. At equal cost, both controls trail LSAR. Figure 6a shows the corresponding seed-resampling ranges.

4.5 Component Analysis

Figure 7a compares seven component variants at the same 1,000-label cap. Removing the trigger ends at 65.1%, 7.3 pp below LSAR's 72.3%, despite spending all 1,000 labels; random acquisition ends at 64.0%, 8.4 pp lower at the same spending. Updating only on audit labels, without additional acquisition, ends at 61.9%. The strict-trigger variant raises the posterior threshold from 0.8 to 0.95. Appendix B.7 specifies every variant. These ablations compare triggering and acquisition rules, but some variants also differ in labels spent and refresh count. The displayed gaps therefore do not isolate a single causal effect; seed-resampling ranges summarize variation across runs.

Panel (b) applies each acquisition rule offline to identical fixed-P8 buffers, so they differ only in which 128 episodes they would label. The rank product selects the largest share of high-score failures at every stage, rising from 25.2% at 0.24 of training to 69.9% at 0.96, against 54.6% for top score and 42.0% for uniform selection at the last stage, when 38.5% of the buffer is such a failure. This offline comparison isolates selection on a common buffer. The held-out audit in Appendix D.6 examines immediate reward precision, while matched controls examine timing at equal cost. Panel (c) shows the LSAR label ledger alongside the periodic ledger.

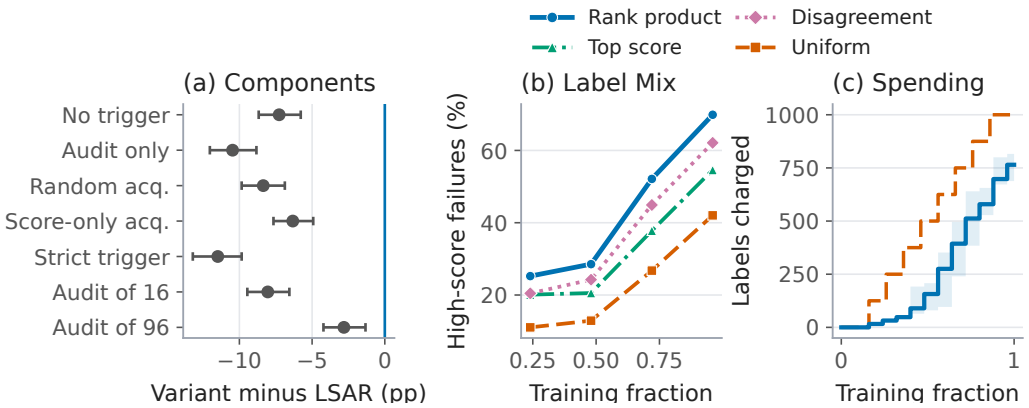

Figure 7: LSAR components on ManiSkill. (a) Final success of each variant minus LSAR with seed-resampling ranges; each variant changes one rule at a cap of 1,000 (Appendix B.7). (b) Share of policy-induced or near-goal failures among 128 episodes selected by each rule from the same fixed-P8 buffers. (c) LSAR cumulative label ledger, mean and seed range, beside the periodic ledger.

4.6 Reward Response and Matched-Cost Controls

The online label cap compares policies with the same maximum annotation budget, but LSAR can stop short of that cap and can refresh at different times. We therefore also compare controls that receive the same number of audit labels at the same 11 boundaries and match LSAR's number of acquisitions and updates for each seed (Appendix D.6). The scheduled top-score control changes which buffered episodes are acquired; the scheduled rank-product control keeps LSAR's acquisition rule but fixes update times before seeing audit outcomes. These comparisons hold the realized label and update counts fixed while allowing policy trajectories and acquired examples to differ.

At matched cost, LSAR finishes at 72.3% on ManiSkill, versus 65.4% for scheduled top score and 67.0% for scheduled rank product, gaps of 6.9 and 5.3 pp. On LIBERO-Long the corresponding values are 40.8%, 32.4%, and 35.2%, gaps of 8.4 and 5.6 pp. The ordinary top-score periodic baseline at the shared 1,000-label cap reaches 66.9% and 33.4%, respectively. Because each control produces a distinct policy trajectory, the matched comparisons support the value of the joint selection and scheduling procedure; they do not assign a separate causal share to each component.

An independent reward audit tests the immediate effect of each refresh without changing the policy. At 23 ManiSkill and 18 LIBERO-Long refresh events, 200 held-out episodes per event are gold-labeled once and scored before and after the reward update. Success among the 40 highest-scored episodes rises from 41.1% to 67.5% on ManiSkill and from 28.5% to 49.9% on LIBERO-Long. The same episodes and labels are used on both sides of each comparison. These 4,600 and 3,600 labels are diagnostic study costs outside the online cap; they are excluded from method-specific label efficiency. The audit establishes improved ranking immediately after refresh, while the final-policy comparisons measure the outcome after continued PPO training.

5 Conclusion

On ManiSkill, fixed P8 loses 9.9 pp from peak to final success as its matched-proxy residual falls. LSAR audits high-scored trajectories and refreshes when gold precision drops. At a 1,000-label cap, it reaches 72.3% final success on ManiSkill and 40.8% on LIBERO-Long, spending a mean of 765 and 637 online gold labels, respectively. The LIBERO-Long gain occurs without a qualifying fixed-reward decline; S8 refresh and transfer beyond these tasks remain untested.

AI USE STATEMENT

AI tools assisted with literature search, manuscript organization and writing edits, discussion of experimental design, and code for figures and analysis. The authors are responsible for the experiments, numerical results, references, analyses, and final manuscript.

ETHICS STATEMENT

The study concerns simulated manipulation tasks. Success under privileged simulator predicates does not establish safe physical deployment, and learned rewards can fail to represent task completion or other safety requirements. The evaluation does not involve human participants, human annotators, or physical robots. Use of public models and benchmarks remains subject to their licenses and terms.

REPRODUCIBILITY STATEMENT

Appendix A defines the selection estimators, the effective-sample-size rule, matching, and the resampling procedure. Appendix B specifies the policy and reward architectures, progress targets, optimizer settings, task splits, success criteria, the audit and acquisition rules, the refresh objective, and every baseline. Appendix C gives the run matrix, interaction and label accounts, statistical endpoints, and failure definitions. The appendices document the run configuration, label accounting, evaluation protocol, and statistical endpoints.

REFERENCES

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In *Advances in Neural Information Processing Systems*, volume 34, pp. 29304–29320, 2021.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025. URL https://arxiv.org/abs/2511.21631.

Ahmad Beirami, Alekh Agarwal, Jonathan Berant, Alexander D'Amour, Jacob Eisenstein, Chirag Nagpal, and Ananda Theertha Suresh. Theoretical guarantees on the best-of-n alignment policy. In *International Conference on Machine Learning*, 2025.

Thomas Coste, Usman Anwar, Robert Kirk, and David Krueger. Reward model ensembles help mitigate overoptimization. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.02743.

Florian E. Dorner, Yatong Chen, André F. Cruz, and Fanny Yang. ROC-n-reroll: How verifier imperfection affects test-time scaling. *arXiv preprint arXiv:2507.12399v3*, 2026. URL https://arxiv.org/abs/2507.12399v3. Version 3, revised August 17, 2026.

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall/CRC, 1994.

Jacob Eisenstein, Chirag Nagpal, Alekh Agarwal, Ahmad Beirami, Alex D'Amour, DJ Dvijotham, Adam Fisch, Katherine Heller, Stephen Pfohl, Deepak Ramachandran, Peter Shaw, and Jonathan Berant. Helping or herding? reward model ensembles mitigate but do not eliminate reward hacking. In *Conference on Language Modeling*, 2024.

João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4):44:1–44:37, 2014.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, 2023. URL https://arxiv.org/abs/2210.10760.

Lin Gui, Cristina Gârbacea, and Victor Veitch. BoNBoN alignment for large language models and the sweetness of best-of-n sampling. In *Advances in Neural Information Processing Systems*, 2024.

Xiao Hu, Jianxiong Li, Xianyuan Zhan, Qing-Shan Jia, and Ya-Qin Zhang. Query-policy misalignment in preference-based reinforcement learning. In *International Conference on Learning Representations*, 2024.

Audrey Huang, Adam Block, Qinghua Liu, Nan Jiang, Akshay Krishnamurthy, and Dylan J. Foster. Is best-of-N the best of them? coverage, scaling, and optimality in inference-time alignment. In *International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 25075–25126, 2025.

Jacek Karwowski, Oliver Hayman, Xingjian Bai, Klaus Kiendlhofer, Charlie Griffin, and Joar Skalse. Goodhart's law in reinforcement learning. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.09144.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025. URL https://arxiv.org/abs/2502.19645.

Augustine Kong, Jun S. Liu, and Wing Hung Wong. Sequential imputations and Bayesian missing data problems. *Journal of the American Statistical Association*, 89(425):278–288, 1994.

Jacky Kwok, Christopher Agia, Rohan Sinha, Matt Foutter, Shulu Li, Ion Stoica, Azalia Mirhoseini, and Marco Pavone. RoboMonkey: Scaling test-time sampling and verification for vision-language-action models. In *Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 3200–3217, 2025.

Kimin Lee, Laura Smith, Anca Dragan, and Pieter Abbeel. B-Pref: Benchmarking preference-based reinforcement learning. In *Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021a.

Kimin Lee, Laura M. Smith, and Pieter Abbeel. PEBBLE: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. In *International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6152–6163, 2021b.

Tony Lee, Andrew Wagenmaker, Karl Pertsch, Percy Liang, Sergey Levine, and Chelsea Finn. RoboReward: General-purpose vision-language reward models for robotics. *arXiv preprint arXiv:2601.00675*, 2026. URL https://arxiv.org/abs/2601.00675.

Haozhan Li, Yuxin Zuo, Jiale Yu, Yuhao Zhang, Zhaohui Yang, Kaiyan Zhang, Xuekai Zhu, Yuchen Zhang, Tianxing Chen, Ganqu Cui, Dehui Wang, Dingxiang Luo, Yuchen Fan, Youbang Sun, Jia Zeng, Jiangmiao Pang, Shanghang Zhang, Yu Wang, Yao Mu, Bowen Zhou, and Ning Ding. SimpleVLA-RL: Scaling VLA training via reinforcement learning. In *International Conference on Learning Representations*, 2026.

Anthony Liang, Yigit Korkmaz, Jiahui Zhang, Minyoung Hwang, Abrar Anwar, Sidhant Kaushik, Aditya Shah, Alex S. Huang, Luke Zettlemoyer, Dieter Fox, Yu Xiang, Anqi Li, Andreea Bobu, Abhishek Gupta, Stephen Tu, Erdem Biyik, and Jesse Zhang. Robometer: Scaling general-purpose robotic reward models via trajectory comparisons. *arXiv preprint arXiv:2603.02115*, 2026. URL https://arxiv.org/abs/2603.02115.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://arxiv.org/abs/2306.03310.

Jijia Liu, Feng Gao, Bingwen Wei, Xinlei Chen, Qingmin Liao, Yi Wu, Chao Yu, and Yu Wang. What can RL bring to VLA generalization? an empirical study. In *Advances in Neural Information Processing Systems*, 2025.

Guanxing Lu, Wenkai Guo, Chubin Zhang, Yuheng Zhou, Haonan Jiang, Zifeng Gao, Yansong Tang, and Ziwei Wang. VLA-RL: Towards masterful and general robotic manipulation with scalable reinforcement learning. *arXiv preprint arXiv:2505.18719*, 2025.

David Manheim and Scott Garrabrant. Categorizing variants of Goodhart's law. *arXiv preprint arXiv:1803.04585*, 2018.

Yuchun Miao, Sen Zhang, Liang Ding, Rong Bao, Lefei Zhang, and Dacheng Tao. InfoRM: Mitigating reward hacking in RLHF via information-theoretic reward modeling. In *Advances in Neural Information Processing Systems*, 2024.

Mitsuhiko Nakamoto, Oier Mees, Aviral Kumar, and Sergey Levine. Steering your generalists: Improving robotic foundation models via value guidance. In *Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 4996–5013, 2025.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022.

Alexander Pan, Kush Bhatia, and Jacob Steinhardt. The effects of reward misspecification: Mapping and mitigating misaligned models. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2201.03544.

Juan Rocamonde, Victoriano Montesinos, Elvis Nava, Ethan Perez, and David Lindner. Vision-language models are zero-shot reward models for reinforcement learning. In *International Conference on Learning Representations*, 2024.

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

Avi Singh, Larry Yang, Kristian Hartikainen, Chelsea Finn, and Sergey Levine. End-to-end robotic reinforcement learning without reward engineering. In *Robotics: Science and Systems*, 2019.

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward hacking. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2209.13085.

Benedikt Stroebl, Sayash Kapoor, and Arvind Narayanan. The limits of inference scaling through resampling. In *International Conference on Learning Representations*, 2026.

Shuhan Tan, Kairan Dou, Yue Zhao, and Philipp Krähenbühl. Interactive post-training for vision-language-action models. *arXiv preprint arXiv:2505.17016*, 2025.

Stone Tao, Fanbo Xiang, Arth Shukla, Yuzhe Qin, Xander Hinrichsen, Xiaodi Yuan, Chen Bao, Xinsong Lin, et al. ManiSkill3: GPU parallelized robotics simulation and rendering for generalizable embodied AI. *arXiv preprint arXiv:2410.00425*, 2024. URL https://arxiv.org/abs/2410.00425.

Lorenz Wolf, Robert Kirk, and Mirco Musolesi. Reward model overoptimisation in iterated RLHF. *arXiv preprint arXiv:2505.18126*, 2025.

Joy Qiping Yang, Salman Salamatian, Ziteng Sun, Ananda Theertha Suresh, and Ahmad Beirami. Asymptotics of language model alignment. In *IEEE International Symposium on Information Theory*, pp. 2027–2032, 2024.

Hongzhi Zang, Mingjie Wei, Si Xu, Yongji Wu, Zhen Guo, Yuanqing Wang, Hao Lin, Peihong Wang, Liangzhi Shi, Yuqing Xie, Zhexuan Xu, Zhihao Liu, Kang Chen, Wenhao Tang, Quanlu Zhang, Weinan Zhang, Chao Yu, and Yu Wang. RLinf-VLA: A unified and efficient framework for reinforcement learning of vision-language-action models. *arXiv preprint arXiv:2510.06710*, 2025. URL https://arxiv.org/abs/2510.06710v3.

Shaopeng Zhai, Qi Zhang, Tianyi Zhang, Fuxian Huang, Haoran Zhang, Ming Zhou, Shengzhe Zhang, Litao Liu, Sixu Lin, and Jiangmiao Pang. A vision-language-action-critic model for robotic real-world reinforcement learning. *arXiv preprint arXiv:2509.15937*, 2025.

Jiahui Zhang, Yusen Luo, Abrar Anwar, Sumedh Anand Sontakke, Joseph J. Lim, Jesse Thomason, Erdem Biyik, and Jesse Zhang. ReWiND: Language-guided rewards teach robot policies without new demonstrations. In *Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 460–488, 2025.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.