# REACHABLE INFORMATION GAIN: TASK-RELEVANT GOAL CURRICULA FOR SPARSE-REWARD LONG-HORIZON REINFORCEMENT LEARNING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Sparse-reward long-horizon reinforcement learning requires agents to discover intermediate behaviors that make distant rewards reachable. Existing methods often select subgoals by novelty, controllability, or reachability, but such signals may not reveal progress toward final success, especially in tasks with distractors. To address this limitation, we introduce Reachable Information Gain (RIG), an evaluation criterion for candidate progress signals in sparse-reward long-horizon reinforcement learning. RIG is motivated by an ideal latent progress criterion, but uses observable multi-step progress over task predicates or goal potentials because latent task progress is not directly identifiable from observations. We instantiate RIG as an online teacher for goal-conditioned off-policy learning. The teacher learns context-dependent utilities from delayed multi-step evidence and supplies curriculum rewards and replay priorities without requiring successful trajectories. Experiments across sparse-reward manipulation, navigation, and discrete-control benchmarks show that RIG improves both sample efficiency and success rate over strong baselines, demonstrating its effectiveness across diverse long-horizon tasks. The code is available at https://anonymous.4open.science/r/RIG.

## 1 INTRODUCTION

Long-horizon decision making with sparse rewards remains a central challenge in reinforcement learning (RL) (Zhu et al., 2026; Yang et al., 2026). In such tasks, an agent may need to execute a long sequence of actions before completing the task and receiving an external reward. Providing intermediate guidance can therefore facilitate exploration and motivate the agent toward potentially useful behaviors before solving the task. Such guidance may take the form of training goals, subgoals, or intrinsic rewards. Existing methods use different evaluation criteria to determine what this guidance should encourage. BEAG (Yoon et al., 2024) and SSE (Hwang et al., 2025) use reachability information to organize subgoal exploration and execution. DEIR (Wan et al., 2023) and ETD (Jiang et al., 2025) quantify episodic novelty using representations learned by a discriminative model and temporal distance. RIDE (Raileanu & Rocktäschel, 2020) and CAI (Seitzer et al., 2021) emphasize changes in learned state representations and action influence, respectively. Although these criteria support extensive exploration, they do not directly measure whether the resulting interactions contribute to accomplishing the current task.

Consider a robotic task in which a robot must pick up an object, move it to a target location, and place it there. During the exploration, the robot may instead interact with a distractor object (Figure 1(a)(b)). Such interaction can still satisfies the criteria of reachability, controllability or action influence, and may therefore be preferentially encouraged by existing exploration criteria. Likewise, interaction with the distractor may also receive novelty rewards, because they are rarely visited in the agent's experience. Nevertheless, these interactions contribute little to the completion of the task. Even when intermediate guidance concerns the task object, its effectiveness can vary with the current state. As illustrated in Figure 1(c), guidance for approaching the object is beneficial before contact is established, whereas encouraging lifting becomes more appropriate once a stable grasp has been achieved. These observations suggest that intermediate guidance should prioritize reachable interactions according to their contribution to task progress under the current state.



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.

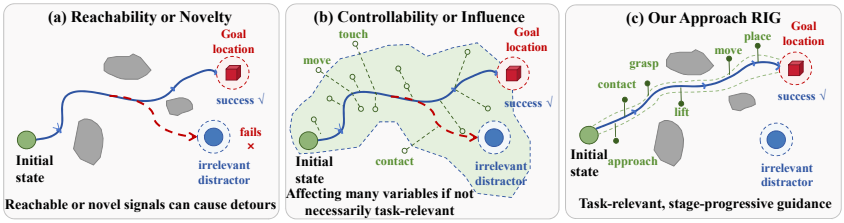



Figure 1: Motivation for RIG: intermediate guidance aligned with task progress. (a) Reachability or novelty may favor interactions with a distractor object. (b) Such interactions may also be controllable or exhibit action influence while contributing little to task completion. (c) Useful guidance depends on the current state. These observations motivate guidance that prioritizes reachable interactions contributing to task progress.

To formalize the selection of intermediate guidance, we consider candidate progress signals: progress measures used to construct auxiliary rewards. We introduce Causal Information Gain (CIG) as an ideal information objective for assessing the task relevance of these signals. Under a short-horizon intervention that fixes a candidate signal as the source of guidance, CIG measures how much the resulting evidence reduces uncertainty about latent task progress. In this sense, a signal is considered informative when intervening on it yields evidence that is predictive of genuine task advancement. We further show that, in general partially observable settings, CIG is not identifiable from observable interaction histories alone.

To address this limitation, we propose **Reachable Information Gain (RIG)** based on a supplied representation of observable task progress. This representation defines an observable counterpart to the latent information objective. For practical evaluation, we associate each candidate with a bounded progress potential and score its largest positive increase from the start of a short trajectory window. By considering the largest improvement over the window, RIG can capture task progress that becomes evident only after a sequence of actions.

We instantiate this idea as ONLINE-RIG, an adaptive teacher for goal-conditioned RL. Given the current observation and task goal, the teacher constructs an observable progress context, estimates the utility of each candidate progress signal from delayed multi-step evidence, and adaptively selects a signal using a UCB-based decision rule. The selected signal is then used to construct an auxiliary reward for the student, while the progress evidence can also be used to prioritize replay. Throughout training, the student remains conditioned on the task goal associated with each training sample, ensuring that auxiliary guidance complements rather than replaces the original task objective.

Our main contributions are summarized as follows: (1) We introduce **Reachable Information Gain** (RIG), an evaluation criterion for candidate progress signals based on observable multi-step evidence of task progress. (2) We establish the non-identifiability of Causal Information Gain (CIG) from observable histories in general partially observable settings and provide conditional guarantees for RIG-based candidate selection. (3) We instantiate RIG as ONLINE-RIG, an adaptive teacher that provides curriculum rewards and replay priorities for agent learning. (4) We evaluate RIG on Panda-Gym, AntMaze, and MiniGrid, demonstrating improved sample efficiency and final performance.

## 2 PROBLEM FORMULATION

### 2.1 GOAL-CONDITIONED SPARSE-REWARD RL

We consider a goal-conditioned Markov decision process $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{G}, P, \mu_0, r^{\text{ext}}, \gamma, \mathcal{H})$ with a finite horizon, where $\mathcal{S}$, $\mathcal{A}$, and $\mathcal{G}$ are the state, action, and goal spaces, $P(s_{t+1} \mid s_t, a_t)$ is the transition distribution, $\mu_0$ is the initial state distribution, $r^{\text{ext}}(s_t, a_t, s_{t+1}, g)$ is the external reward for goal $g$, $\gamma \in (0, 1]$ is the discount factor, and $\mathcal{H}$ is the episode horizon. The agent uses a goal-conditioned policy $a_t \sim \pi_\theta(\cdot \mid s_t, g)$ and maximizes the expected return $\sum_{t=0}^{\mathcal{H}-1} r^{\text{ext}}$ over $\mathcal{H}$. We focus on sparse-reward tasks, where $r^{\text{ext}}$ provides informative feedback only when task is completed.

### 2.2 CANDIDATE UTILITY

Intermediate guidance can supplement sparse rewards by supporting exploration and learning before task completion. Motivated by goal-conditioned learning (Pitis et al., 2020; Diaz-Bone et al., 2026), we assume access to a finite set $\mathcal{Z}$ of candidate progress signals and focus on evaluating their utility rather than generating them. Given state $s_t$, goal $g$, and candidate $z \in \mathcal{Z}$, let $U(z \mid s_t, g)$ denote the utility of $z$ for learning the final task. Existing approaches use evaluation criteria such as reachability, novelty, and action influence to guide exploration. These criteria alone do not establish whether the resulting interactions advance the current task. We therefore seek a candidate utility that reflects whether the resulting guidance encourages reachable interactions that provide evidence of task progress in the current state.

## 3 REACHABLE INFORMATION GAIN

### 3.1 IDEAL INFORMATION GAIN OVER LATENT TASK PROGRESS

Let $h_t = (s_0, a_0, \ldots, s_t)$ denote the interaction history at time $t$. Let $H \leq \mathcal{H} - t$ be a local evidence horizon, and $\mathbf{X}_t^H = (X_t^\star, \ldots, X_{t+H}^\star)$ denote the latent variables over this horizon. These variables represent task-relevant factors that govern progress toward the desired goal, but their values and semantics are not assumed to be available to the learner. For a candidate signal $z \in \mathcal{Z}$, $\mathrm{do}(z)$ denotes an idealized intervention that fixes $z$ as the signal being practiced for the next $H$ steps, without changing the environment dynamics. Let $E_{z,t}^H$ denote the resulting evidence, such as state changes or progress events. All information quantities are evaluated under the candidate-conditioned rollout distribution $P_{z,t} := P(\cdot \mid h_t, \mathrm{do}(z))$.

We define the *Causal Information Gain* (CIG) of candidate $z$ as

$$\mathrm{CIG}(z \mid h_t) = I_{P_{z,t}}\big(\mathbf{X}_t^H; E_{z,t}^H\big)\,, \tag{1}$$

where $I_P(U; V)$ denotes the mutual information between random variables $U$ and $V$ under distribution $P$. CIG measures the informativeness of implementation evidence for latent task progress under idealized interventions. Exact CIG is generally unavailable during learning because $\mathbf{X}_t^H$ is not observed and evaluating each candidate would require its candidate-conditioned rollout distribution. This limitation becomes fundamental under partial observability: even complete observable histories need not identify the latent CIG.

**Theorem 1** (Non-identifiability of exact CIG)**.** *There exist two goal-conditioned partially observable environments that induce the same distribution over observable histories for every policy, but assign different CIG values to the same candidate at some history. Consequently, exact CIG over latent task-progress variables is not identifiable from observable histories alone.*

Proof and the equivalent posterior-divergence form of Eq. 1 are given in Appendix A.

### 3.2 OBSERVABLE PROGRESS AND OPERATIONAL RIG

To obtain a usable objective, we replace latent task progress with an observable progress representation. For task goal $g$, let

$$Y_t = \rho(s_t, g) = \big(Y_t^{(1)}, \ldots, Y_t^{(K)}\big) \in [0,1]^K, \mathbf{Y}_t^H = (Y_t, \ldots, Y_{t+H}), \tag{2}$$

where $\rho : \mathcal{S} \times \mathcal{G} \to [0,1]^K$ is a task-progress extractor. Each component $Y_t^{(i)}$ is a binary predicate or a normalized continuous feature associated with observable task progress. Unlike $\mathbf{X}_t^H$, the trajectory $\mathbf{Y}_t^H$ is available from observed states and does not require a known ordered task decomposition. The information-theoretic form of Reachable Information Gain is

$$\mathrm{RIG}_{\mathrm{MI}}(z \mid h_t) = I_{P_{z,t}}\big(\mathbf{Y}_t^H; E_{z,t}^H\big)\,, \tag{3}$$

which measures how informative the reachable consequences of candidate $z$ are about observable task progress.

**Proposition 1** (Observable information bound)**.** *If, under $P_{z,t}$, $\mathbf{Y}_t^H \perp\!\!\!\perp E_{z,t}^H \mid \mathbf{X}_t^H$, then*

$$\mathrm{RIG}_{\mathrm{MI}}(z \mid h_t) \leq \mathrm{CIG}(z \mid h_t). \tag{4}$$

Under this condition, $\mathbf{Y}_t^H - \mathbf{X}_t^H - E_{z,t}^H$ forms a Markov chain under $P_{z,t}$, and the result follows from the data processing inequality. The proof is provided in Appendix A.

Directly estimating Eq. 3 is still difficult in continuous domains. We therefore associate each candidate $z$ with a bounded observable potential $\Phi_z : \mathcal{S} \times \mathcal{G} \to [0, 1]$, where $\Phi_z(s, g)$ measures the progress associated with candidate $z$ at state $s$, given task goal $g$. Given an $H$-step trajectory segment, we define the operational multi-step progress score

$$u_t^H(z; g) = \max_{1 \le k \le H} [\Phi_z(s_{t+k}, g) - \Phi_z(s_t, g)]_+ , \tag{5}$$

where $[x]_+ = \max(x, 0)$. The score measures the largest positive potential increase relative to the start of the trajectory window, capturing progress that may occur before the endpoint and later disappear. The operational score is not a direct estimator of Eq. equation 3. Under a monotone calibration condition between observable progress and ideal CIG, however, its candidate ranking approximately preserves the ideal ranking. The formal guarantee is provided in Appendix A.

**Controlled CIG-RIG diagnostic.** To illustrate the connection between ideal CIG and observable RIG, we use a small tabular setting which the latent progress variables and candidate-conditioned distributions are known, allowing both CIG and RIG to be computed exactly. As shown in Figure 2, RIG closely preserves the CIG ranking when the observable progress representation is informative. As task information is progressively corrupted, the RIG score range contracts and becomes uninformative under complete corruption. This illustrates the role of the representation assumption underlying the observable surrogate. Full numerical results and additional diagnostics are provided in Appendix F.

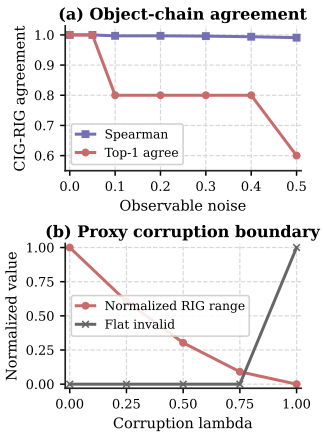



Figure 2: Exact tabular CIG-RIG diagnostics.

## 4 Online-RIG

### 4.1 Method Overview

To turn the sample-level evidence in Eq. 5 into adaptive learning guidance, we introduce *Online-RIG*. Online-RIG learns which candidate progress signals are useful in different progress contexts and uses these utilities to guide an underlying goal-conditioned learner.

Online-RIG consists of a *student* and a *teacher*. The student follows a goal-conditioned policy $\pi_\theta(a \mid s, g)$, where $g$ is the training goal for the current sample, either the original environment goal or a relabeled goal when goal relabeling is used. All goal-dependent quantities in that update are evaluated with respect to the same $g$. The teacher maps the current observable progress to a context $c_t$, selects a candidate signal $z_t \in \mathcal{Z}$, and evaluates it using the delayed multi-step evidence $u_t^H(z_t; g)$.

The candidate $z_t$ does not replace $g$ or enter the student policy. Instead, it specifies which observable progress signal is emphasized during the current update. The resulting evidence updates the teacher's context-dependent utility estimate, and the selected candidate provides the curriculum reward for the student. Thus, Online-RIG uses reachable trajectory evidence as an operational surrogate for the candidate-practice intervention underlying RIG, rather than directly estimating $P(\cdot \mid h_t, \mathrm{do}(z))$. For replay-based learners, episode priorities are computed separately using the original episode goal, as described in Section 4.5.

### 4.2 Observable Progress Context

The usefulness of an intermediate signal depends on the progress already achieved. We capture this dependence by mapping the observable progress representation $Y_t$ in Eq. 2 to a finite context:

$$c_t = C(Y_t) = C(\rho(s_t, g)) \in \mathcal{C}, \tag{6}$$

where $C : [0,1]^K \to \mathcal{C}$ is a deterministic context map and $\mathcal{C}$ is a finite set of observable progress contexts. The map $C$ may encode discrete progress features directly or discretize continuous features.

The contexts distinguish situations in which different candidates may be useful, but they do not impose a known transition chain. Unless stated otherwise, every candidate in $\mathcal{Z}$ is available in every context, so Online-RIG does not require a hand-coded context-specific candidate mask.

### 4.3 Online Candidate Evaluation

For each context-candidate pair $(c, z)$, define the current operational utility

$$\mu(c, z) = \mathbb{E}\big[u_t^H(z; g) \mid c_t = c\big], \tag{7}$$

where the expectation is taken under the current student and replay distribution. This utility measures the expected reachable progress associated with candidate $z$ in context $c$. Since both the replay distribution and the student policy change during learning, $\mu(c, z)$ is estimated online.

The teacher maintains an estimate $\widehat{\mu}(c, z)$ and a count $N(c, z)$ for every pair:

$$\widehat{\mu}(c, z) = 0, \qquad N(c, z) = 0, \qquad \forall (c, z) \in \mathcal{C} \times \mathcal{Z}. \tag{8}$$

Let $N(c) = \sum_{z \in \mathcal{Z}} N(c, z)$ be the total number of candidate evaluations in context $c$. During warm-up, candidates are sampled uniformly. Afterwards, the teacher computes

$$A(c, z) = \widehat{\mu}(c, z) + \lambda_{\text{ucb}} \sqrt{\frac{\log(N(c) + 1)}{N(c, z) + 1}}, \tag{9}$$

where $\lambda_{\text{ucb}} \geq 0$ controls exploration. The teacher selects $z_t$ using an $\epsilon$-greedy policy over $A(c_t, \cdot)$.

After selecting $z_t$ and retrieving the following $H$-step segment from replay, the teacher computes $u_t^H(z_t; g)$ and updates

$$N(c_t, z_t) \leftarrow N(c_t, z_t) + 1, \tag{10}$$

$$\widehat{\mu}(c_t, z_t) \leftarrow (1 - \eta)\widehat{\mu}(c_t, z_t) + \eta u_t^H(z_t; g), \tag{11}$$

where $\eta \in (0, 1]$ is the teacher learning rate. Estimates for unselected pairs remain unchanged. The constant step size allows the teacher to track nonstationary candidate utilities as the student improves.

### 4.4 Curriculum Reward

The delayed utility $u_t^H$ evaluates candidate usefulness, but the student requires a transition-level reward for temporal difference learning. For the selected candidate $z_t$, define the one-step potential increment

$$\Delta_t(z_t; g) = \Phi_{z_t}(s_{t+1}, g) - \Phi_{z_t}(s_t, g). \tag{12}$$

Since $\Phi_z(s, g) \in [0, 1]$, the increment lies in $[-1, 1]$. Online-RIG supplies the curriculum reward

$$r_t^{\text{RIG}} = \alpha \, \Delta_t(z_t; g), \tag{13}$$

where $\alpha \geq 0$ controls the strength of the auxiliary signal. The reward used for the student update is

$$r_t^{\text{train}} = r^{\text{ext}}(s_t, a_t, s_{t+1}, g) + r_t^{\text{RIG}}. \tag{14}$$

When goal relabeling is applied, both terms are recomputed using the relabeled training goal.

The multi-step evidence $u_t^H$ and the one-step increment $\Delta_t$ play different roles. The former updates the teacher's estimate of candidate usefulness, while the latter provides a local training signal for the selected candidate. Since $z_t$ depends on the observable context and the teacher's learned utilities, $r_t^{\text{RIG}}$ is an adaptive curriculum reward rather than a fixed potential-based shaping term (Ng et al., 1999). It is therefore not intended to provide a policy-invariance guarantee, and all methods are evaluated using the original external success criterion.

### 4.5 Replay Prioritization

For replay-based learners, Online-RIG can additionally prioritize episodes containing useful progress evidence (Schaul et al., 2015). Let $e \in \mathcal{B}$ denote an episode with its original task goal $g_e$. When its priority is refreshed, we recompute

$$c_t^e = C(\rho(s_t, g_e))$$

for each timestep and greedily select a priority candidate $z_t^{\text{prio}} = \arg\max_{z \in \mathcal{Z}} A(c_t^e, z)$ using the current teacher. These candidates are recomputed rather than permanently stored in replay.

Let $\mathcal{C}(e) = \{c_t^e : t \in e\}$ denote progress contexts appearing in $e$. We define the episode priority as

$$p(e) = \varepsilon_p + \frac{1}{|\mathcal{C}(e)|} \sum_{c \in \mathcal{C}(e)} \max_{\substack{t \in e \\ c_t^e = c}} u_t^H(z_t^{\text{prio}}; g_e), \tag{15}$$

where $\varepsilon_p > 0$ ensures nonzero sampling probability. Averaging across contexts prevents frequently visited contexts from dominating the priority, while the inner maximum retains the strongest progress evidence. Episodes are sampled proportionally to $p(e)^\kappa$, where $\kappa \geq 0$ controls the prioritization strength.

After an episode is sampled, transition sampling and HER relabeling proceed as usual. The candidate for the student update is selected separately using the relabeled training goal. Thus, replay prioritization changes only the training-data distribution without introduces additional reward.

## 5 Experiments

Our experiments aim to answer the following questions. RQ1: Does Online-RIG improve sample efficiency and final task performance across sparse-reward *manipulation, navigation, and discrete-control tasks*? RQ2: Can RIG distinguish task-progressive signals from novel or reachable but task-irrelevant distractor signals? RQ3: Does multi-step evidence outperform one-step or endpoint-only progress? RQ4: Can Online-RIG learn useful context-dependent candidate selection from unordered signals? RQ5: How do curriculum rewards and replay prioritization contribute to performance, and how robust is RIG to imperfect observable progress representations?

### 5.1 Experimental Setup

**Benchmarks.** We evaluate Online-RIG on three sparse-reward benchmark families: goal-conditioned robotic manipulation, long-horizon navigation, and partially observable discrete control. For **Panda-Gym** (Gallouédec et al., 2021), we use Push, Flip, Slide, and PickAndPlace, together with **four PickAndPlace distractor variants** that introduce a controllable but task-irrelevant object. The four variants include matched, biased, path, and random, corresponding to symmetric, closer, route-blocking, and randomly placed distractors, respectively. For **AntMaze** (Lee et al., 2022; Hwang et al., 2025), we use U-maze, Bottleneck-maze, DoubleBottleneck-maze and AntKeyChest. For **MiniGrid** (Chevalier-Boisvert et al., 2023), we use KC-S3R3, KC-S5R3, OM-2Dlh, and OM-1Q, where KC and OM denote KeyCorridor and ObstructedMaze, respectively. Detailed experimental configurations are provided in Appendix B.

**Baselines.** We compare Online-RIG with the following baselines: HER (Andrychowicz et al., 2017), AMIGo (Campero et al., 2020), DISCOVER (Diaz-Bone et al., 2026), RIDE (Raileanu & Rocktäschel, 2020), DEIR (Wan et al., 2023), ETD (Jiang et al., 2025), CAI (Seitzer et al., 2021), BEAG (Yoon et al., 2024) and SSE (Hwang et al., 2025). Each baseline is evaluated only on compatible benchmark families, due to differences in state, action, and goal spaces across these domains. For each task, all methods use the same training budget, evaluation frequency, and random seeds. Baseline implementations and hyperparameters are provided in Appendix C.

**Implementation.** For Panda-Gym and AntMaze, Online-RIG is implemented on top of DDPG with HER. For MiniGrid, it is on top of PPO. Unless otherwise stated, Online-RIG uses evidence horizon $H = 15$, curriculum weight $\alpha = 1.0$, teacher learning rate $\eta = 0.05$, UCB coefficient $\lambda_{\text{ucb}} = 0.25$, and replay-priority exponent $\kappa = 1.0$. All experiments are conducted on 2 × A800 GPUs. Network architectures and the experiments on the selection of hyperparameters are provided in Appendix C.

**Evaluation Metrics.** We report success rate as the main metric, and mean episode return for MiniGrid. For aggregate comparison, we report final success rate (Final SR), best success rate (Best SR), area under the success curve (AUC), and the first timestep reaching $80\%$ success, denoted $t_{0.8}$. We report $95\%$ bootstrap confidence intervals and paired bootstrap tests for the main AUC and $t_{0.8}$ comparisons in Appendix D.

### 5.2 Overall Performance

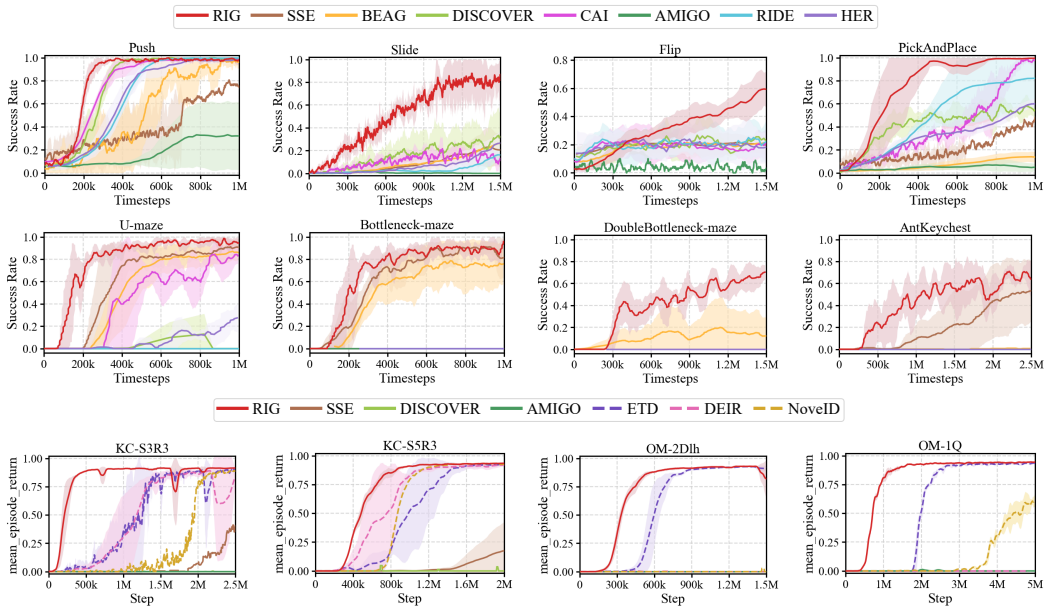



Figure 3: Learning curves on three benchmarks. Each subplot corresponds to one task. Solid curves show the mean success rate over six random seeds, and shaded regions show one standard deviation.

To answer **RQ1**, we compare Online-RIG with compatible baselines on 12 tasks across Panda-Gym, AntMaze, and MiniGrid. As shown in Figure 3, Online-RIG achieves the comparable final performance on all tasks and generally reaches high success or return with fewer environment interactions.

**Panda-Gym.** Online-RIG learns faster on all manipulation tasks, reaching near-perfect success earliest on **Push** and **PickAndPlace**, and achieving higher final success on **Slide** and **Flip**. These gains are most pronounced in tasks requiring coordinated multi-stage interactions.

**AntMaze.** Online-RIG consistently improves navigation, converging faster on **U-maze** and **Bottleneck-maze** and maintaining clear gains on **DoubleBottleneck-maze** and **AntKeyChest**. The results highlight its benefit in tasks with bottlenecks and context-dependent intermediate objectives.

**MiniGrid.** Online-RIG also learns fastest in partially observable discrete tasks, reaching higher returns earlier on the **KeyCorridor** and **ObstructedMaze** variants. This shows that RIG is not limited to continuous goal spaces.

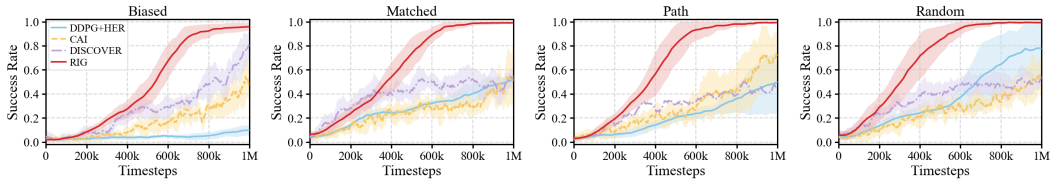



Figure 4: Performance of RIG and baselines under four different distractor settings.

### 5.3 Task Relevance under Distractors

To answer **RQ2**, we evaluate Online-RIG under matched, biased, path, and random distractor layouts. As shown in Figure 4, Online-RIG learns faster and reaches near-perfect success across all layouts, while baselines are more sensitive to distractor placement. Figure 5 further reports the distractor top-1 rate, defined as the fraction of anchor-state and stage pairs whose highest-ranked candidate is distractor-related. Online-RIG keeps this rate near $0.10$, with the largest gap in the biased layout where the closer distractor attracts achievability-based and influence-based guidance. These results show that RIG improves learning by selecting signals aligned with long-horizon task progress, rather than generic novelty or controllability.

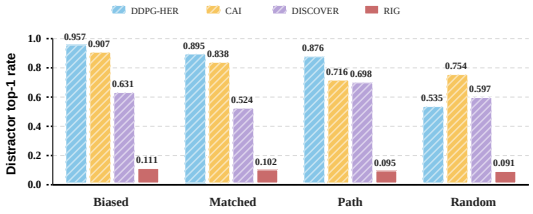



Figure 5: Distractor top-1 rate under four PickAndPlace distractor layouts. Lower values indicate better task relevance.

### 5.4 Effect of Multi-Step Evidence

To answer **RQ3**, we ablate the teacher update target on PickAndPlace Biased-Distractor. Table 1 shows that one-step and endpoint-only evidence fail to reach $0.8$ success, while window-based multi-step evidence succeeds. The max-window score in Eq. 5 performs best, reaching $0.975$ final success and $t_{0.8} = 0.98$M. This supports using multi-step maximum evidence because manipulation progress can occur transiently within the window and disappear by the endpoint. Additional diagnostics are provided in Appendix E.

Table 1: Multi-step evidence aggregation ablation on PickAndPlace Biased-Distractor.

| Evidence | Final SR | Best SR | AUC | $t_{0.8}$ |
|---|---|---|---|---|
| One-step, $H$=1 | 0.513 | 0.725 | 0.218 | – |
| Endpoint, $H$=15 | 0.338 | 0.588 | 0.198 | – |
| Sum/mean, $H$=15 | 0.963 | 0.975 | 0.393 | 1.24M |
| Max, $H$=15 | **0.975** | **1.000** | **0.496** | **0.98M** |

### 5.5 Context-Dependent Candidate Selection

To answer **RQ4**, we test whether performance depends on learned context-dependent selection rather than merely exposing the agent to dense intermediate signals. We use the same unordered symbolic candidate set in progress contexts and compare uniform selection with the Online-RIG context-conditioned bandit. Table 2 shows that uniform selection performs poorly, while Online-RIG reaches $0.988$ final success and $t_{0.8} = 0.84$M. Thus, the gain comes from learning which signal is useful in each context, not from the candidate set alone. More detailed analysis are provided in Appendix G.

Table 2: Context-dependent candidate selection with the same unordered symbolic candidates.

| Selection | Final SR | Best SR | AUC | $t_{0.8}$ |
|---|---|---|---|---|
| Uniform | 0.325 | 0.413 | 0.144 | – |
| Online-RIG | **0.988** | **1.000** | **0.517** | **0.84M** |

### 5.6 Component Ablations and Robustness

To answer **RQ5**, we ablate reward guidance, replay prioritization, and candidate sources. Table 3 shows that RIG-P alone solves the biased distractor task without changing rewards, improving final success from $0.325$ to $1.000$. The curriculum reward is also strong, and combining it with RIG-P gives the highest AUC and fastest $t_{0.8}$. This shows that RIG progress evidence can guide both reward shaping and replay selection.

Table 3: Component ablation on biased PickAndPlace-Distractor. Rwd. indicates Online-RIG curriculum reward, and RIG-P indicates RIG replay prioritization.

| Method | Rwd. | RIG-P | Final SR | AUC | $t_{0.8}$ |
|---|---|---|---|---|---|
| HER | – | – | 0.325 | 0.143 | – |
| RIG-P only | – | ✓ | 1.000 | 0.429 | 1.18M |
| RIG reward | ✓ | – | 0.988 | 0.581 | 0.70M |
| RIG full | ✓ | ✓ | **1.000** | **0.605** | **0.67M** |

Table 4: Structured feature candidates reduce manual design while preserving strong performance.

| Candidate source | Design | Final SR | Best SR | AUC |
|---|---|---|---|---|
| Symbolic predicates | symbolic | 0.925 | 0.963 | **0.421** |
| Object-sym. feature | feature | **0.963** | **0.963** | 0.338 |
| Achieved feature | replay feat | 0.638 | 0.875 | 0.272 |
| Achieved-goal xyz | position | 0.150 | 0.250 | 0.078 |

Table 4 evaluates candidate-source robustness in the same biased distractor environment. Symbolic predicates provide the strongest compact prior, but Online-RIG remains effective with less manually specified progress features: object-symmetric features match symbolic final success, and replay-derived achieved features reach high best success. In contrast, achieved-goal xyz alone is much weaker, showing that RIG does not benefit from arbitrary dense signals, and the candidate representation must retain manipulation-progress information. Additional landmark and random-feature controls are reported in Appendix H.

## 6 Related Work

**Intermediate goals guidance.** HER relabels failed trajectories with achieved goals, while Skew-Fit and MEGA promote goal-space coverage or achieved-goal entropy (Nair et al., 2018; Pong et al., 2019; Pitis et al., 2020). AMIGo generates adversarial intrinsic goals, and DISCOVER balances achievability, novelty, and target relevance (Florensa et al., 2018; Campero et al., 2020; Diaz-Bone et al., 2026). Hierarchical and graph-based methods instead improve subgoal feasibility through adjacency constraints, landmarks, graph search, adaptive grids, or reliable execution (Kim et al., 2021; Lee et al., 2022; Yoon et al., 2024; Hwang et al., 2025). These approaches primarily evaluate goals through achieved states, value estimates, coverage, or reachability. RIG instead evaluates whether practicing a candidate produces observable multi-step evidence of final task progress.

**Intrinsic motivation and causal exploration.** ICM and RND reward prediction error (Pathak et al., 2017; Burda et al., 2018), while RIDE, NovelD, E3B, DEIR, ETD encourage exploration through representation change, episodic novelty, elliptical bonuses, discriminative rewards, or temporal distance (Raileanu & Rocktäschel, 2020; Zhang et al., 2021; Henaff et al., 2022; Wan et al., 2023; Jiang et al., 2025). CAI rewards causally influential interactions (Seitzer et al., 2021). Although these signals promote novelty or controllability, they favor task-irrelevant changes. RIG makes task relevance by scoring candidate consequences according to observable progress toward final success.

## 7 Conclusion

We presented RIG, an evaluation criterion for candidate progress signals in sparse-reward long-horizon RL. RIG evaluates candidates using observable multi-step evidence of task progress, with theoretical connections to Causal Information Gain (CIG) under explicit assumptions. We instantiated this criterion as Online-RIG, an adaptive teacher that learns context-dependent candidate utilities and supports learning through auxiliary rewards and optional replay prioritization. Experiments on Panda-Gym, AntMaze, and MiniGrid demonstrate improved sample efficiency and final performance, including in tasks with controllable distractors.

### AI USE STATEMENT

AI tools were used to assist with literature search, manuscript polishing and proofreading, and assessment of the theoretical rigor of the manuscript.

### ETHICS STATEMENT

This work does not involve human participants, sensitive personal information, or applications with direct societal impact. All experiments are conducted on publicly available benchmark datasets that are widely adopted by the research community.

### REPRODUCIBILITY STATEMENT

To facilitate reproducibility and promote transparency, we publicly release our code at `https://anonymous.4open.science/r/RIG`. Additional details for reproducing our results are provided in Appendix C.

## REFERENCES

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *Advances in neural information processing systems*, 30, 2017.

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

Andres Campero, Roberta Raileanu, Heinrich Küttler, Joshua B Tenenbaum, Tim Rocktäschel, and Edward Grefenstette. Learning with amigo: Adversarially motivated intrinsic goals. *arXiv preprint arXiv:2006.12122*, 2020.

Maxime Chevalier-Boisvert, Bolun Dai, Mark Towers, Rodrigo De Lazcano, Lucas Willems, Salem Lahlou, Suman Pal, Pablo Samuel Castro, and Jordan Terry. Minigrid & miniworld: Modular & customizable reinforcement learning environments for goal-oriented tasks. 2023.

Leander Diaz-Bone, Marco Bagatella, Jonas Hübotter, and Andreas Krause. Discover: Automated curricula for sparse-reward reinforcement learning. *Advances in Neural Information Processing Systems*, 38:21863–21896, 2026.

Carlos Florensa, David Held, Xinyang Geng, and Pieter Abbeel. Automatic goal generation for reinforcement learning agents. In *International Conference on Machine Learning*, pp. 1515–1528. PMLR, 2018.

Quentin Gallouédec, Nicolas Cazin, Emmanuel Dellandréa, and Liming Chen. panda-gym: Open-source goal-conditioned environments for robotic learning. *arXiv preprint arXiv:2106.13687*, 2021.

Mikael Henaff, Roberta Raileanu, Minqi Jiang, and Tim Rocktäschel. Exploration via elliptical episodic bonuses. *Advances in Neural Information Processing Systems*, 35:37631–37646, 2022.

Jaebak Hwang, Sanghyeon Lee, Jeongmo Kim, and Seungyul Han. Strict subgoal execution: Reliable long-horizon planning in hierarchical reinforcement learning. *arXiv preprint arXiv:2506.21039*, 2025.

Yuhua Jiang, Qihan Liu, Yiqin Yang, Xiaoteng Ma, Dianyu Zhong, Hao Hu, Jun Yang, Bin Liang, Bo Xu, Chongjie Zhang, et al. Episodic novelty through temporal distance. *arXiv preprint arXiv:2501.15418*, 2025.

Junsu Kim, Younggyo Seo, and Jinwoo Shin. Landmark-guided subgoal generation in hierarchical reinforcement learning. *Advances in neural information processing systems*, 34:28336–28349, 2021.

Seungjae Lee, Jigang Kim, Inkyu Jang, and H Jin Kim. Dhrl: A graph-based approach for long-horizon and sparse hierarchical reinforcement learning. *Advances in neural information processing systems*, 35:13668–13678, 2022.

Ashvin V Nair, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine. Visual reinforcement learning with imagined goals. *Advances in neural information processing systems*, 31, 2018.

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

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pp. 2778–2787. PMLR, 2017.

Silviu Pitis, Harris Chan, Stephen Zhao, Bradly Stadie, and Jimmy Ba. Maximum entropy gain exploration for long horizon multi-goal reinforcement learning. In *International conference on machine learning*, pp. 7750–7761. PMLR, 2020.

Vitchyr H Pong, Murtaza Dalal, Steven Lin, Ashvin Nair, Shikhar Bahl, and Sergey Levine. Skew-fit: State-covering self-supervised reinforcement learning. *arXiv preprint arXiv:1903.03698*, 2019.

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

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.

Maximilian Seitzer, Bernhard Schölkopf, and Georg Martius. Causal influence detection for improving efficiency in reinforcement learning. *Advances in Neural Information Processing Systems*, 34: 22905–22918, 2021.

Shanchuan Wan, Yujin Tang, Yingtao Tian, and Tomoyuki Kaneko. Deir: efficient and robust exploration through discriminative-model-based episodic intrinsic rewards. *arXiv preprint arXiv:2304.10770*, 2023.

Jiarui Yang, Bin Zhu, Jingjing Chen, and Yu-Gang Jiang. Actor-critic for continuous action chunks: A reinforcement learning framework for long-horizon robotic manipulation with sparse reward. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18692–18700, 2026.

Youngsik Yoon, Gangbok Lee, Sungsoo Ahn, and Jungseul Ok. Breadth-first exploration on adaptive grid for reinforcement learning. In *Forty-first International Conference on Machine Learning*, 2024.

Tianjun Zhang, Huazhe Xu, Xiaolong Wang, Yi Wu, Kurt Keutzer, Joseph E Gonzalez, and Yuandong Tian. Noveld: A simple yet effective exploration criterion. *Advances in Neural Information Processing Systems*, 34:25217–25230, 2021.

Yijie Zhu, Rui Shao, Ziyang Liu, Jie He, Jizhihui Liu, Jiuru Wang, and Zitong Yu. H-gar: A hierarchical interaction framework via goal-driven observation-action refinement for robotic manipulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18882–18890, 2026.

# A Additional Theory and Proofs

Throughout this appendix, we fix a history $h_t$ and a candidate $z \in \mathcal{Z}$. All distributions and information quantities are evaluated under the candidate-conditioned rollout distribution

$$P_{z,t} := P(\cdot \mid h_t, \mathrm{do}(z)). \tag{16}$$

For compactness, we use $\mathbf{X} := \mathbf{X}_t^H$, $\mathbf{Y} := \mathbf{Y}_t^H$, $E := E_{z,t}^H$.

## A.1 Equivalent Posterior-Divergence Form of CIG

The mutual-information definition in Eq. 1 admits the following posterior-update interpretation.

**Proposition 2** (Expected posterior divergence)**.** *Assume that the relevant regular conditional distributions exist and that the mutual information is finite. Then*

$$\mathrm{CIG}(z \mid h_t) = \mathbb{E}_{e \sim P_{z,t}(E)} \left[ D_{\mathrm{KL}} \left( P_{z,t}(\mathbf{X} \mid E = e) \,\|\, P_{z,t}(\mathbf{X}) \right) \right]. \tag{17}$$

*Proof.* By the definition of mutual information,

$$\begin{aligned} I_{P_{z,t}}(\mathbf{X}; E) &= \mathbb{E}_{(\mathbf{x},e) \sim P_{z,t}} \left[ \log \frac{P_{z,t}(\mathbf{x}, e)}{P_{z,t}(\mathbf{x}) P_{z,t}(e)} \right] \\ &= \mathbb{E}_{(\mathbf{x},e) \sim P_{z,t}} \left[ \log \frac{P_{z,t}(\mathbf{x} \mid e)}{P_{z,t}(\mathbf{x})} \right] \\ &= \mathbb{E}_{e \sim P_{z,t}(E)} \left[ D_{\mathrm{KL}} \left( P_{z,t}(\mathbf{X} \mid E = e) \,\|\, P_{z,t}(\mathbf{X}) \right) \right]. \end{aligned} \tag{18}$$

Combining Eq. 18 with Eq. 1 proves the result. □

Whenever the corresponding entropies are finite, the same quantity can equivalently be written as

$$\mathrm{CIG}(z \mid h_t) = H_{P_{z,t}}(\mathbf{X}) - H_{P_{z,t}}(\mathbf{X} \mid E). \tag{19}$$

Thus, CIG measures the expected reduction in uncertainty about the latent progress trajectory after observing evidence generated while practicing $z$. Because $\mathbf{X}_t^H$ contains progress variables after candidate practice begins, its reference distribution is $P(\mathbf{X}_t^H \mid h_t, \mathrm{do}(z))$, rather than the observational distribution $P(\mathbf{X}_t^H \mid h_t)$.

## A.2 Proof of Theorem 1

*Proof.* We construct two partially observable environments that induce the same observable-history distribution under every policy but assign different CIG values to the same candidate.

Consider a one-step candidate-practice problem at a fixed observable history $\bar{h}_t$. After $\mathrm{do}(z)$, the environment produces binary evidence $E \in \{0, 1\}$ and a binary terminal-success outcome $S \in \{0, 1\}$. The external terminal reward may be any deterministic function of $S$.

Let $q \in (0, \frac{1}{2})$ and consider the observable joint distribution

$$\begin{aligned} P(E = 0, S = 0) = P(E = 1, S = 1) &= \frac{1 - q}{2}, \\ P(E = 0, S = 1) = P(E = 1, S = 0) &= \frac{q}{2}. \end{aligned} \tag{20}$$

Both variables have uniform marginals and agree with probability $1 - q$.

We now construct two latent models that induce this same observable distribution. In environment $\mathcal{M}$, let $X^\star \sim \mathrm{Bernoulli}(\frac{1}{2})$ and $N \sim \mathrm{Bernoulli}(q)$ independently, and define

$$E = X^\star, \qquad S = X^\star \oplus N, \tag{21}$$

where $\oplus$ denotes exclusive OR. The latent variable $X^\star$ is task relevant because it determines the distribution of terminal success.

In environment $\mathcal{M}'$, let $\widetilde{X}^\star \sim \mathrm{Bernoulli}(\frac{1}{2})$ and $N \sim \mathrm{Bernoulli}(q)$ independently, and define

$$S = \widetilde{X}^\star, \qquad E = \widetilde{X}^\star \oplus N. \tag{22}$$

Here, $\widetilde{X}^\star$ is task relevant because it determines terminal success directly.

In both environments, $E = S$ exactly when $N = 0$. Hence, Eqs. 21 and 22 induce the same observable distribution in Eq. 20. Since the construction is independent of subsequent actions, the two environments induce the same observable-history distribution under every policy.

To match the notation of the main text, set $H = 1$ and let all but one component of $\mathbf{X}_t^H$ be deterministic. The only nonconstant component is $X^\star$ in $\mathcal{M}$ and $\widetilde{X}^\star$ in $\mathcal{M}'$. Deterministic components contribute no mutual information.

In $\mathcal{M}$, the evidence reveals the latent variable exactly:

$$\mathrm{CIG}_{\mathcal{M}}(z \mid \bar{h}_t) = I(X^\star; E) = 1 \text{ bit}. \tag{23}$$

In $\mathcal{M}'$, $E$ is obtained from $\widetilde{X}^\star$ through a binary symmetric channel with crossover probability $q$, giving

$$\mathrm{CIG}_{\mathcal{M}'}(z \mid \bar{h}_t) = I(\widetilde{X}^\star; E) = 1 - h_{\mathrm{b}}(q), \tag{24}$$

where

$$h_{\mathrm{b}}(q) = -q \log_2 q - (1-q) \log_2 (1-q)$$

is the binary entropy function. Since $q \in (0, \frac{1}{2})$, we have $h_{\mathrm{b}}(q) > 0$, and therefore

$$\mathrm{CIG}_{\mathcal{M}}(z \mid \bar{h}_t) \neq \mathrm{CIG}_{\mathcal{M}'}(z \mid \bar{h}_t). \tag{25}$$

Thus, the same observable-history distribution is compatible with different exact latent CIG values. No estimator based only on observable histories can recover the correct CIG in both environments, which proves the theorem. □

The theorem concerns exact CIG over an unspecified latent task-progress representation. It does not imply that useful intermediate signals cannot be learned from observable experience. Instead, it shows that relating observable progress to latent task progress requires additional representational assumptions, motivating the surrogate introduced in Section 3.2.

### A.3 Proof of Proposition 1

*Proof.* Fix $h_t$ and $z$ and evaluate all quantities under $P_{z,t}$, $\mathbf{Y} - \mathbf{X} - E$ forms a Markov chain. Using the chain rule for mutual information,

$$I_{P_{z,t}}(\mathbf{X}, \mathbf{Y}; E) = I_{P_{z,t}}(\mathbf{X}; E) + I_{P_{z,t}}(\mathbf{Y}; E \mid \mathbf{X}) = I_{P_{z,t}}(\mathbf{X}; E),$$
$$I_{P_{z,t}}(\mathbf{X}, \mathbf{Y}; E) = I_{P_{z,t}}(\mathbf{Y}; E) + I_{P_{z,t}}(\mathbf{X}; E \mid \mathbf{Y}) \geq I_{P_{z,t}}(\mathbf{Y}; E).$$

Therefore,

$$I_{P_{z,t}}(\mathbf{Y}; E) \leq I_{P_{z,t}}(\mathbf{X}; E). \tag{26}$$

Substituting Eqs. 1 and 3 into Eq. 26 yields Eq. 4, completing the proof. □

The conditional-independence condition is a substantive assumption about the relation among latent progress, observable progress, and candidate evidence. It does not follow merely because $\mathbf{Y}_t^H$ is observable. For example, if $E_{z,t}^H$ contains state information not represented by $\mathbf{X}_t^H$, conditioning on $\mathbf{X}_t^H$ may not remove all dependence between $\mathbf{Y}_t^H$ and $E_{z,t}^H$. In that case, Proposition 1 does not apply without enriching the latent representation or redefining the evidence.

### A.4 Operational RIG and Approximate Candidate Preservation

To make the connection between this operational score and the ideal CIG explicit, define the population operational utility

$$R_z(h_t) = \mathbb{E}_{P_{z,t}}\left[u_t^H(z; g)\right]. \tag{27}$$

The following sufficient condition states when maximizing an empirical operational score preserves the ideal candidate choice up to bounded distortion.

**Assumption 1** (Monotone operational calibration)**.** *For a fixed history $h_t$, let $C_z = \mathrm{CIG}(z \mid h_t)$ and let $R_z(h_t)$ be defined by Eq. 27. There exist a strictly increasing function $f : \mathbb{R}_{\geq 0} \to \mathbb{R}$ and a constant $\delta \geq 0$ such that*

$$\big|R_z(h_t) - f(C_z)\big| \leq \delta, \qquad \forall z \in \mathcal{Z}. \tag{28}$$

**Proposition 3** (Approximate candidate preservation)**.** *Assume Assumption 1. Let $\widehat{R}_z$ be an empirical or online estimate of $R_z(h_t)$ satisfying*

$$\max_{z \in \mathcal{Z}} \big|\widehat{R}_z - R_z(h_t)\big| \leq \epsilon. \tag{29}$$

*Let $z^\star \in \arg\max_{z \in \mathcal{Z}} C_z$ and $\widehat{z} \in \arg\max_{z \in \mathcal{Z}} \widehat{R}_z$. Then*

$$f(C_{z^\star}) - f(C_{\widehat{z}}) \leq 2(\delta + \epsilon). \tag{30}$$

*If there exists $m > 0$ such that $f(c_1) - f(c_2) \geq m(c_1 - c_2)$ for all $c_1 \geq c_2$ on the candidate range, then*

$$C_{z^\star} - C_{\widehat{z}} \leq \frac{2(\delta + \epsilon)}{m}. \tag{31}$$

Proposition 3 states that operational RIG preserves the ideal candidate up to bounded regret when the observable potential is calibrated to task-progress information and the online estimate is accurate.

**Interpretation of Assumption 1.** Assumption 1 is a calibration condition rather than a consequence of observability or boundedness alone. It states the additional property required for an operational progress score to approximately preserve the ordering induced by ideal CIG. Proposition 3 therefore provides a stability guarantee conditional on the quality of the observable representation, rather than showing that arbitrary potentials $\Phi_z$ estimate mutual information. The exact-tabular experiments in Appendix F examine this condition in settings where both quantities can be enumerated.

*Proof.* For fixed $h_t$, write $C_z := \mathrm{CIG}(z \mid h_t)$ and $R_z := R_z(h_t)$. Let $z^\star \in \arg\max_{z \in \mathcal{Z}} C_z$ and $\widehat{z} \in \arg\max_{z \in \mathcal{Z}} \widehat{R}_z$.

By the definition of $\widehat{z}$, $\widehat{R}_{\widehat{z}} \geq \widehat{R}_{z^\star}$. Together with the uniform estimation bound in Eq. 29, this implies

$$\begin{aligned} R_{\widehat{z}} + \epsilon \geq \widehat{R}_{\widehat{z}} \geq \widehat{R}_{z^\star} \geq R_{z^\star} - \epsilon, \\ R_{z^\star} - R_{\widehat{z}} \leq 2\epsilon. \end{aligned} \tag{32}$$

The calibration condition in Eq. 28 then yields

$$f(C_{z^\star}) \leq R_{z^\star} + \delta \quad \leq R_{\widehat{z}} + 2\epsilon + \delta \leq f(C_{\widehat{z}}) + 2(\delta + \epsilon). \tag{33}$$

Under the lower-slope condition in Proposition 3, we further obtain

$$m\big(C_{z^\star} - C_{\widehat{z}}\big) \leq f(C_{z^\star}) - f(C_{\widehat{z}}) \leq 2(\delta + \epsilon). \tag{34}$$

Together, these inequalities establish the bounds in Eqs. 30 and 31. □

**Corollary 1** (Pairwise ranking preservation)**.** *Under Assumption 1 and Eq. 29, for any $z_1, z_2 \in \mathcal{Z}$,*

$$f(C_{z_1}) - f(C_{z_2}) > 2(\delta + \epsilon) \quad \Longrightarrow \quad \widehat{R}_{z_1} > \widehat{R}_{z_2}. \tag{35}$$

*Proof.* The calibration and estimation bounds imply

$$\begin{aligned} \widehat{R}_{z_1} \geq f(C_{z_1}) - \delta - \epsilon, \\ \widehat{R}_{z_2} \leq f(C_{z_2}) + \delta + \epsilon. \end{aligned} \tag{36}$$

Condition 35 makes the first lower bound strictly larger than the second upper bound. □

**Corollary 2** (Top-one recovery)**.** *Suppose $z^\star$ is the unique CIG-optimal candidate and define its calibrated margin as*

$$\Delta_f := f(C_{z^\star}) - \max_{z \neq z^\star} f(C_z). \tag{37}$$

*If*

$$\Delta_f > 2(\delta + \epsilon), \tag{38}$$

*then $\widehat{z} = z^\star$.*

*Proof.* Condition 38 satisfies Corollary 1 for $z^\star$ and every $z \neq z^\star$. Hence $\widehat{R}_{z^\star} > \widehat{R}_z$ for every competing candidate, so $\widehat{z} = z^\star$. □

When $\delta = \epsilon = 0$, every strictly increasing calibration function $f$ yields identical CIG and operational-RIG rankings. More generally, candidate recovery depends on the quality of the observable progress representation, the accuracy of the online estimates, and the separation between candidate utilities. These results provide sufficient conditions rather than claiming that arbitrary observable potentials preserve the ideal CIG ranking.

### A.5 From History-Level Utility to Context-Level Estimation

For a context $c \in \mathcal{C}$, define the context-averaged operational utility

$$\bar{R}_c(z) := \mathbb{E}\left[R_z(h_t) \mid C(\rho(s_t, g)) = c\right]. \tag{39}$$

The Online-RIG teacher estimates $\bar{R}_c(z)$ under the current student and replay distribution.

For a history $h_t$ mapped to context $c$, decompose

$$|\widehat{\mu}(c,z) - R_z(h_t)| \leq \underbrace{\left|R_z(h_t) - \bar{R}_c(z)\right|}_{\text{context aggregation error}} + \underbrace{\left|\bar{R}_c(z) - \mu_{\mathcal{D}}(c,z)\right|}_{\text{distribution-shift error}} + \underbrace{\left|\mu_{\mathcal{D}}(c,z) - \widehat{\mu}(c,z)\right|}_{\text{online estimation error}}, \tag{40}$$

where $\mu_{\mathcal{D}}(c,z)$ denotes the candidate utility under the current replay and behavior distribution $\mathcal{D}$.

**Proposition 4** (Context-level candidate preservation)**.** *Fix $h_t$ with context $c_t = c$. Suppose the three terms in Eq. 40 are uniformly bounded by $\epsilon_{\text{ctx}}$, $\epsilon_{\text{shift}}$, and $\epsilon_{\text{est}}$, respectively. Under Assumption 1, selecting*

$$\widehat{z}_c \in \arg\max_{z \in \mathcal{Z}} \widehat{\mu}(c,z)$$

*satisfies*

$$f(C_{z^\star}) - f(C_{\widehat{z}_c}) \leq 2\left(\delta + \epsilon_{\text{ctx}} + \epsilon_{\text{shift}} + \epsilon_{\text{est}}\right). \tag{41}$$

## B Benchmarks Details

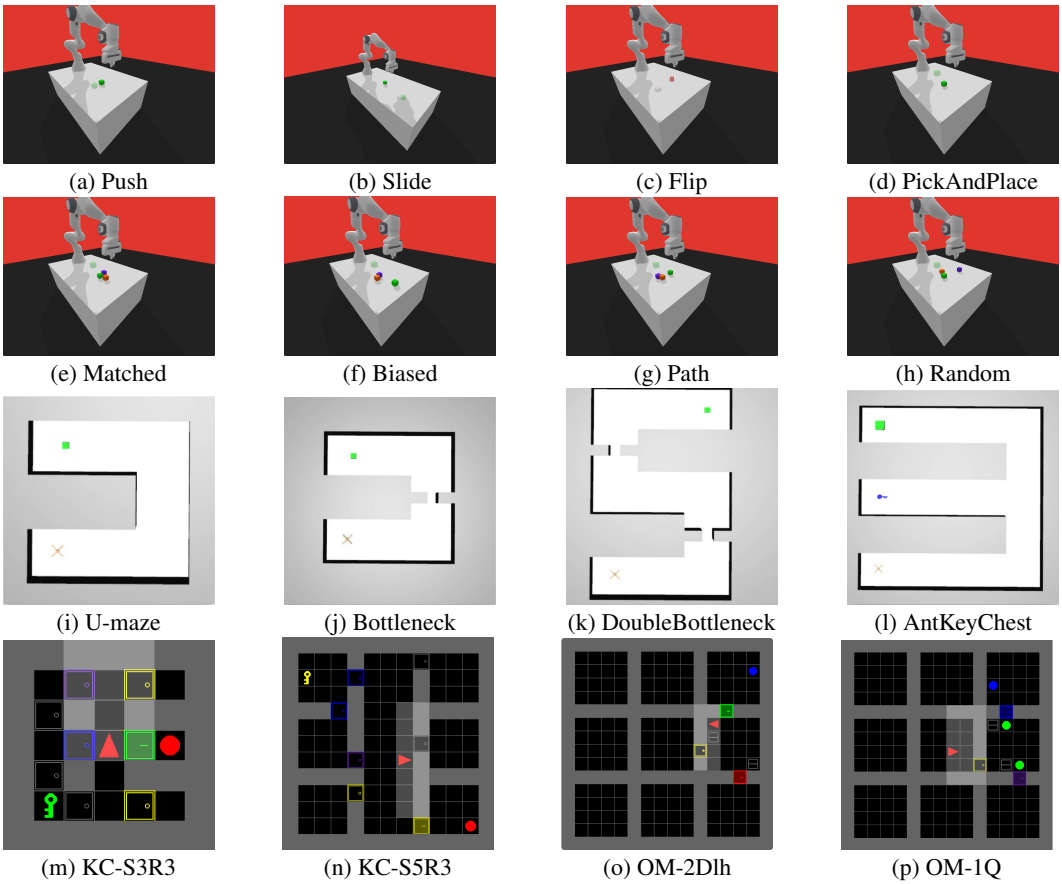
We evaluate Online-RIG on three benchmark families covering continuous robotic manipulation, long-horizon navigation, and partially observable discrete control. The selected tasks vary in state and action spaces, episode horizons, and sources of long-horizon difficulty. Additionally, we constructed four distractor environments for pickandplace task. This appendix describes the task objectives, sparse-reward conditions, and distractor constructions. The environments used in our experiments are visualized in Figure 6. Training budgets and algorithm-specific configurations are provided in Appendix C.

### B.1 Panda-Gym Manipulation Tasks

Panda-Gym provides goal-conditioned robotic manipulation environments with a simulated Franka Emika Panda arm. The agent receives continuous Cartesian control commands for the end effector, together with a gripper command when required by the task. Observations contain the robot state, object state, and desired goal. The external reward is sparse and indicates whether the task-specific goal condition has been satisfied.

We consider the following four tasks.

- **Push**. The robot must contact a movable cube and push it across the table to a target position. Successful behavior requires the agent to approach the correct object, establish contact, and maintain a suitable pushing direction until the object reaches the goal region.
- **Slide**. The robot must strike a rollable object so that it slides toward a distant target. Unlike Push, the object may continue moving after contact is lost. The task therefore requires the agent to coordinate contact position, direction, and velocity so that the resulting object trajectory reaches the target region.



(a) Push (b) Slide (c) Flip (d) PickAndPlace
(e) Matched (f) Biased (g) Path (h) Random
(i) U-maze (j) Bottleneck (k) DoubleBottleneck (l) AntKeyChest
(m) KC-S3R3 (n) KC-S5R3 (o) OM-2Dlh (p) OM-1Q

Figure 6: Renderings of the sixteen benchmark environments used in this work. (a)-(d) are Panda-Gym tasks, (e)-(h) are PickAndPlace distractor variants, (i)-(l) are AntMaze tasks, and (m)-(p) are MiniGrid tasks.

- **Flip**. The robot must manipulate an object from its initial orientation to the desired target orientation. The task requires contact that produces the correct rotational motion rather than merely translating the object. Success is determined by whether the final orientation lies within the task tolerance.
- **PickAndPlace**. The robot must grasp an object, lift it from the table, move it toward the desired location, and release it in the goal region. This task contains a clear sequence of prerequisite behaviors, including approach, contact, grasping, lifting, transport, and placement, making it particularly suitable for evaluating stage-progressive intermediate guidance.

#### B.1.1 PickAndPlace Distractor Variants

To evaluate whether a method distinguishes task-progressive signals from controllable but irrelevant behavior, we augment PickAndPlace with additional movable objects. Only the task object contributes to external success. However, the distractors can still be approached, touched, grasped or moved, and thus can receive high scores under the criteria of accessibility, novelty or influence.

We consider four distractor layouts.

- **Matched**. The task object and distractor are placed in approximately symmetric configurations relative to the robot. This layout reduces simple geometric biases and tests whether the guidance mechanism can identify the task-relevant object when objects are similarly reachable.

- **Biased**. The distractor is initialized closer to the robot than the task object. It is therefore easier to approach and manipulate, creating a strong bias toward immediately reachable but task-irrelevant interactions.
- **Path**. The distractor is placed near or along the direct manipulation route between the robot, task object, and goal region. This layout tests whether the agent can avoid an irrelevant object that is likely to be encountered during task execution.
- **Random**. The distractor position is sampled from the valid workspace at the beginning of each episode. This setting introduces variation in distractor reachability and tests robustness across changing spatial configurations.

Across all four variants, the desired goal and sparse external reward are defined only for the task object. The distractor does not alter the success condition.

### B.2 AntMaze Navigation Tasks

The AntMaze benchmarks require a quadruped agent with continuous locomotion controls to navigate through maze layouts toward a target location. The observation contains the agent state and goal information, while the external reward is sparse and depends on reaching the desired goal. Narrow corridors, bottlenecks, and distant objectives make direct exploration difficult because locally reachable regions may not lead toward final success.

We use the following four environments.

- **U-maze.** The agent starts on one side of a U-shaped obstacle and must navigate around the enclosing wall to reach a goal that is geometrically close but not directly accessible. The task tests whether intermediate guidance can avoid locally attractive directions that lead into the obstacle.
- **Bottleneck-maze**. The start and goal regions are separated by a narrow passage. The agent must first reach and traverse the bottleneck before moving toward the final goal. Exploration that does not identify the passage is unlikely to obtain external reward.
- **DoubleBottleneck-maze**. This environment contains multiple constrained passages that must be crossed in sequence. Compared with Bottleneck-maze, it requires a longer chain of intermediate navigation decisions and increases the chance that the agent becomes trapped in locally reachable but unproductive regions.
- **AntKeyChest**. The agent must first reach a key location and then navigate to the chest or final target. Goal completion is conditioned on key acquisition, so directly reaching the final region without the prerequisite event is insufficient. This environment tests whether the learned curriculum adapts to context-dependent intermediate objectives.

For the first three tasks, success is determined by reaching the final target region. AntKeyChest additionally includes the key-acquisition condition before final completion.

### B.3 MiniGrid Tasks

MiniGrid provides discrete, partially observable environments that combine navigation with object interaction. At each step, the agent receives a local egocentric observation and selects from a discrete action set, including movement, turning, and task-dependent interactions. The time-dependent success reward is $r = 1 - 0.9t/T$, where $t$ is the completion step and $T$ is the maximum episode length. The reward is zero when the task is not completed. Earlier completion therefore produces a larger episode return.

We evaluate two KeyCorridor and two ObstructedMaze tasks.

- **KC-S3R3**. KeyCorridor requires the agent to locate and pick up a key, unlock a door, and reach the target region to pick it up. The notation 'S' denotes the number of rooms, while 'R' signifies the size of each room. The S3R3 configuration uses a comparatively compact layout, but still requires a sequence of navigation and interaction events that cannot be inferred from the terminal reward alone.

- **KC-S5R3**. This task follows the same key-door-goal structure as KC-S3R3 but uses a larger layout. The increased navigation distance and number of irrelevant locations make discovery of the prerequisite key and door interactions more difficult.
- **OM-2Dlh**. ObstructedMaze combines navigation with interaction involving movable obstacles and objects. It is the most challenging environment in the minigrid, requiring agents to first unlock a key inside a box before opening a locked door and collecting the task object. 'Dl' indicates the number of locked doors, 'h' denotes that keys are hidden within boxes, an obstructed door by a ball is indicated by 'b', and 'Q' represents how many of the nine original quadrants on the map will contain both doors and keys. In the 2Dlh setting, there are two locked doors, with two keys hidden in two separate boxes. Due to partial observability and long interaction chains, randomly discovering successful paths becomes difficult, resulting in lower success probabilities.
- **OM-1Q**. The 1Q variant is even more challenging. It not only includes two locked doors and two keys in the box, but also two balls that need to be moved aside before the task can be completed. It is similar to OM-2Dlhb, but has a larger maximum number of steps.

Together, these MiniGrid environments test whether Online-RIG can use observable progress signals in discrete and partially observable domains, rather than relying on continuous goal-space geometry.

# C Baselines and Implementation Details

We compare Online-RIG with the following baselines: HER (Andrychowicz et al., 2017), AMIGo (Campero et al., 2020), DISCOVER (Diaz-Bone et al., 2026), RIDE (Raileanu & Rocktäschel, 2020), DEIR (Wan et al., 2023), ETD (Jiang et al., 2025), CAI (Seitzer et al., 2021), BEAG (Yoon et al., 2024) and SSE (Hwang et al., 2025). Each baseline is evaluated only on compatible benchmark families, due to differences in state, action, and goal spaces across these domains. For each task, all methods use the same training budget, evaluation frequency, and random seeds.

## C.1 Baeline Details

We compare Online-RIG with the following baselines:

**HER.** Hindsight Experience Replay (HER) (Andrychowicz et al., 2017) relabels failed trajectories with achieved goals, enabling off-policy goal-conditioned learning from sparse rewards. We use DDPG+HER as the base learner for Panda-Gym and AntMaze.

**AMIGo.** AMIGo (Campero et al., 2020) trains a goal-generating teacher that proposes goals for student policy and is rewarded when the goals are neither too easy nor impossible. We include it as a learned curriculum baseline where its goal generation interface is compatible with the task.

**DISCOVER.** DISCOVER (Diaz-Bone et al., 2026) constructs and expands a set of reachable goals to guide exploration in goal-conditioned environments.It serves as a benchmark for goal curriculum based on reachability and coverage scope.

**RIDE.** RIDE (Raileanu & Rocktäschel, 2020) is an intrinsic motivation method that rewards changes in state embeddings, encouraging controllable exploration. We use it as a task-agnostic exploration baseline, especially in MiniGrid.

**DEIR.** DEIR (Wan et al., 2023) improves intrinsic exploration by combining episodic novelty with learned representations that reduce noisy or irrelevant novelty signals. We compare against it as a stronger intrinsic motivation baseline in discrete-control tasks.

**ETD.** ETD (Jiang et al., 2025) is an episodic exploration method that prioritizes temporally distinct or informative experience within an episode. It provides a recent episodic novelty baseline for MiniGrid-style sparse reward tasks.

**CAI.** Causal Action Influence (CAI) (Seitzer et al., 2021) scores objects or state variables by how much actions causally influence them. We use CAI to test whether influence alone is sufficient for task-relevant guidance under distractors.

**BEAG.** BEAG (Yoon et al., 2024) is a graph-based exploration method for long-horizon goal-conditioned navigation. We include it as an AntMaze baseline that explicitly reasons about breadth-first exploration over reachable regions.

**SSE.** Strict Subgoal Execution (SSE) (Hwang et al., 2025) is a hierarchical method that constructs subgoal graphs and enforces reliable execution of selected subgoals. We compare against SSE because of its excellent performance in long-term sequence navigation tasks of bottleneck-type.

### C.2 Implementations

For Panda-Gym and AntMaze, Online-RIG is implemented on top of DDPG with HER. For MiniGrid, Online-RIG is implemented on top of PPO. The teacher is used only during training to select intermediate progress signals and, when enabled, replay priorities. At evaluation time, the teacher is disabled and the student policy is evaluated using the original task objective. Unless otherwise stated, Online-RIG uses the hyperparameters in Table 5.

Table 5: Default Online-RIG hyperparameters.

| Hyperparameter | Value |
|---|---|
| Evidence horizon $H$ | 15 |
| Curriculum reward weight $\alpha$ | 1.0 |
| Teacher learning rate $\eta$ | 0.05 |
| UCB coefficient $\lambda_{\mathrm{ucb}}$ | 0.25 |
| Candidate selection | $\epsilon$-greedy UCB bandit |
| Replay-priority exponent $\kappa$ | 1.0 |
| Replay-priority mode | stage-balanced, when enabled |
| Evaluation teacher | disabled |
| Evaluation reward | original sparse task reward |

Since the three domains differ in observation structure, action space, and goal representation, each baseline is evaluated only on compatible tasks. When a baseline was originally proposed and released for a given benchmark family, we use the official implementation and the hyperparameters reported in the corresponding paper or code release. For methods ported to our benchmark suite, we keep the base learner, training budget, evaluation frequency, and random seeds matched to Online-RIG.

### C.3 Domain-Specific Settings

For Panda-Gym, all methods are trained with the same DDPG+HER backbone and task budget. Distractor experiments use four layouts: matched, biased, path, and random. The distractor layouts change only the placement of task-irrelevant objects. And the task goal, success criterion, and evaluation protocol are unchanged.

For AntMaze, Online-RIG follows the same training interface as the SSE AntMaze codebase. For U-maze and Bottleneck, the total training budget is 1M environment steps. For the longer AntMaze tasks, including DoubleBottleneck and KeyChest, the budget is 5M steps. For KeyChest task, we use the task-specific horizon and exploration settings from the SSE codebase, since these tasks require longer episodes and broader initial graph coverage. Although these runs were allocated a maximum budget of 5M steps, Figure 3 displays 1 M/1.5M/2.5M steps, in order to present our results more clearly in comparison with the baselines.

For MiniGrid, RIG is instantiated with a PPO student using partial $7 \times 7$ observations. The symbolic teacher operates online during rollout collection: it maintains the current finite-state curriculum stage, selects a valid subgoal before interaction. After each environment transition, oracle symbolic predicates are used to compute the curriculum reward and update the $H$-step option credit. The resulting one-step curriculum reward is stored in the PPO rollout buffer and added before generalized advantage estimation and PPO updates.

We use a rollout length and batch size of 512, $\gamma = 0.99$, GAE parameter $\lambda = 0.95$, and learning rate $3 \times 10^{-4}$. The student uses a lightweight MiniGrid CNN encoder, which performed better than

Table 6: Observable progress interface used in the Panda-Gym experiments. Progress features, contexts, candidates, and thresholds are fixed before training. Online-RIG learns only their context-dependent utilities and does not use a manually specified candidate order.

| Task | Observable progress information | Progress contexts | Candidate signals |
|---|---|---|---|
| Push | Gripper-object proximity, object contact, and object-goal displacement | Approach, contact, transport, and completion contexts | Approach, contact, move, and place |
| Slide | Gripper-object proximity, contact, and object progress toward the target | Approach, contact, motion, and completion contexts | Approach, contact, move, and place |
| Flip | Object-relative interaction features and progress toward the desired orientation | Pre-contact, interaction, rotation-progress, and completion contexts | Approach, contact, rotate, and complete |
| PickAndPlace | Gripper–object proximity, contact, grasp proxy, object height, object–goal distance, placement, and release proxies | Far, near-contact, grasped, lifted, near-goal, and placed/released | Approach, contact, grasp, lift, move, place, and release |
| PickAndPlace-Distractor | The PickAndPlace features evaluated for the task and distractor objects; external success depends only on the task object | The same observable contexts as PickAndPlace | Task-object and distractor-object counterparts of the same candidate library |

the larger IMPALA CNN in our implementation. Since PPO does not use an episode replay buffer, the replay-prioritization mechanism is disabled. Thus, the MiniGrid results labeled RIG include stage-conditioned symbolic subgoal selection and RIG curriculum rewards, but not RIG-P.

### C.4 Manual Knowledge Interface: Panda-Gym Example

We use Panda-Gym as a representative example to clarify the manual task knowledge required by Online-RIG. Before training, we specify a compact observable progress interface consisting of a progress representation $\rho$, a finite context set $\mathcal{C}$ with context map $C$, a candidate set $\mathcal{Z}$, and bounded candidate potentials $\Phi_z$. These quantities are computed from the standard object-centric environment state.

This interface does not specify which candidate should be used in each context. In particular, Online-RIG does not use a hand-written stage-to-candidate ordering or a context-specific valid-candidate mask in the no-mask experiments. Instead, the teacher learns the context-dependent utility of every candidate from multi-step trajectory evidence. Thus, the manually provided knowledge defines the observable vocabulary of progress, while the curriculum over that vocabulary is learned online.

Table 6 summarizes the fixed progress interface used in the Panda-Gym experiments. The PickAndPlace interface is described in detail below because it contains the richest sequence of manipulation events.

**PickAndPlace Progress Interface**. Let $p_g$, $p_o$, and $p^\star$ denote the Cartesian positions of the gripper, manipulated object, and desired goal, respectively. Let $p_o^z$ denote the vertical coordinate of the object and let $w$ denote the gripper width. We use $\mathrm{clip}(x, 0, 1) = \min(1, \max(0, x))$, and $\mathbb{I}[\cdot]$ denotes the indicator function.

The candidate set is $\mathcal{Z}_{\mathrm{PnP}} = \{\texttt{approach}, \texttt{contact}, \texttt{grasp}, \texttt{lift}, \texttt{move}, \texttt{place}, \texttt{release}\}$.

For each candidate, the corresponding observable potential is defined as

$$\begin{aligned}
\Phi_{\text{approach}}(s,g) &= \text{clip}\left(1 - \frac{\|p_g - p_o\|_2}{0.12},\, 0, 1\right), \\
\Phi_{\text{contact}}(s,g) &= \mathbb{I}\left[\|p_g - p_o\|_2 < 0.065\right], \\
\Phi_{\text{grasp}}(s,g) &= \Phi_{\text{contact}}(s,g)\, \mathbb{I}[w < 0.055], \\
\Phi_{\text{lift}}(s,g) &= \text{clip}\left(\frac{p_o^z - 0.03}{0.12},\, 0, 1\right), \\
\Phi_{\text{move}}(s,g) &= \exp\left(-\frac{\|p_o - p^\star\|_2^2}{2(0.12)^2}\right) \Phi_{\text{lift}}(s,g), \\
\Phi_{\text{place}}(s,g) &= \exp\left(-\frac{\|p_o - p^\star\|_2^2}{2(0.06)^2}\right), \\
\Phi_{\text{release}}(s,g) &= \Phi_{\text{place}}(s,g)\, \mathbb{I}[w > 0.065].
\end{aligned} \tag{42}$$

The observable context map uses thresholded versions of the same progress quantities. Contexts are assigned in the following precedence order: `placed/released` → `near-goal` → `lifted` → `grasped` → `near-contact` → `far`. The precedence rule ensures that each state is assigned to one context when several predicates are simultaneously satisfied. It defines a coarse observable progress partition, not a candidate curriculum. Every candidate in $\mathcal{Z}_{\text{PnP}}$ remains available in every context, and the teacher learns the utility of each context-candidate pair online.

**Distractor Interface and Task Identity**. For PickAndPlace-Distractor, the same candidate potentials are evaluated separately for the task object and each distractor object. This produces object-symmetric candidate counterparts, such as task-object contact and distractor-object contact. However, the desired goal, sparse external reward, and environment success condition are defined only with respect to the task object.

The environment interface therefore provides the identity of the goal-relevant object. The distractor experiments do not evaluate unsupervised discovery of object roles from raw observations. Instead, they test whether candidate selection remains aligned with the specified task goal when another object is equally or more reachable and controllable.

Accordingly, "without a hand-written curriculum" means that Online-RIG does not use a manually specified candidate order, stage transition graph, or context-specific candidate mask. It does not mean that the method automatically discovers the task-relevant object, predicate vocabulary, or progress representation.

**Other Panda-Gym Tasks**. Push, Slide, and Flip use the same separation between a fixed observable progress interface and a learned curriculum. Their interfaces retain only task-relevant quantities, such as gripper-object proximity, contact, object-goal progress, or orientation progress. Online-RIG then learns which candidate signal is useful in each observable context. Exact thresholds and environment-specific potential definitions are listed in the released configuration files.

### C.5 Hyperparameter Selection

We tune the main reward-scale hyperparameter on PickAndPlace and then keep the selected values fixed in the main experiments. Table 7 shows the sweep over the curriculum reward weight $\alpha$. Low weights improve final success but learn more slowly, while $\alpha = 0.8$ and $\alpha = 1.0$ form a stable high-performing plateau, with $\alpha = 1.0$ performing the best. We use $\alpha = 1.0$ as the default in the main experiments.

For the evidence horizon, we use $H = 15$. This choice is supported by the multi-step evidence ablation in Table 1: one-step evidence is too local, endpoint-only evidence misses transient progress, and the $H = 15$ max-window target gives the best sample efficiency. We keep $H$ fixed rather than tuning it separately for each benchmark. Other teacher parameters, including $\eta$, $\lambda_{\text{ucb}}$, and replay-priority exponent $\kappa$, are fixed pilot defaults across experiments.

Table 7: Curriculum reward weight sweep on PickAndPlace. Higher $\alpha$ improves sample efficiency up to a stable plateau, $\alpha = 1.0$ yields the best result.

| $\alpha$ | Final SR | AUC | $t_{0.5}$ | $t_{0.8}$ |
|---|---|---|---|---|
| 0.2 | 1.000 | 0.651 | 555k | 750k |
| 0.3 | 0.987 | 0.688 | 567k | 653k |
| 0.6 | 1.000 | 0.766 | 360k | 500k |
| 0.8 | 1.000 | 0.803 | **313k** | **400k** |
| 1.0 | 1.000 | **0.816** | **313k** | **400k** |

**Algorithm 1** Online-RIG with HER and Replay Prioritization

**Require:** student policy $\pi_\theta$, replay buffer $\mathcal{B}$, candidate set $\mathcal{Z}$, progress extractor $\rho$, context map $C$, potentials $\{\Phi_z\}_{z\in\mathcal{Z}}$

Initialize teacher estimates $\widehat{\mu}(c, z)$ and counts $N(c, z)$
**while** training **do**
  Collect an episode $e$ using $\pi_\theta(\cdot \mid s, g_e)$ with the environment goal $g_e$
  Store the raw episode $e$ in $\mathcal{B}$
  $p(e) \leftarrow \textsc{ScoreEpisode}(e)$
  **for** each student update **do**
    Sample episode $e \sim P_{\mathcal{B}}(e) \propto p(e)^\kappa$
    Sample transition index $t$ and retrieve $(s_t, a_t, s_{t+1}, \ldots, s_{t+H})$ from $e$
    Sample a training goal $g$ using HER (or retain the original goal $g_e$)
    Recompute $r_t^{\text{ext}} = r^{\text{ext}}(s_t, a_t, s_{t+1}, g)$
    Compute observable progress $Y_t = \rho(s_t, g)$ and context $c_t = C(Y_t)$
    Select training candidate $z_t^{\text{train}}$ using the $\epsilon$-greedy teacher over $A(c_t, \cdot)$
    Compute delayed evidence

$$u_t^H(z_t^{\text{train}}; g) = \max_{1\le k\le H} [\Phi_{z_t^{\text{train}}}(s_{t+k}, g) - \Phi_{z_t^{\text{train}}}(s_t, g)]_+$$

    Increment $N(c_t, z_t^{\text{train}})$ and update

$$\widehat{\mu}(c_t, z_t^{\text{train}}) \leftarrow (1-\eta)\widehat{\mu}(c_t, z_t^{\text{train}}) + \eta u_t^H(z_t^{\text{train}}; g)$$

    Compute one-step candidate progress

$$\Delta_t = \Phi_{z_t^{\text{train}}}(s_{t+1}, g) - \Phi_{z_t^{\text{train}}}(s_t, g)$$

    Form student reward

$$r_t^{\text{train}} = r_t^{\text{ext}} + \alpha\Delta_t$$

    Update the student using $(s_t, a_t, r_t^{\text{train}}, s_{t+1}, g)$
    Refresh the cached episode priority $p(e) \leftarrow \textsc{ScoreEpisode}(e)$
  **end for**
**end while**

### C.6 End-to-End Training Procedure

Algorithm 1 summarizes the complete training procedure for replay-based Online-RIG. It clarifies the order of replay sampling, HER relabeling, context computation, candidate selection, teacher update, curriculum-reward construction, and student update. Episode priorities are refreshed separately using the original episode goal and candidates reselected by the current teacher, rather than reusing the candidates selected for HER-based student updates.

## D Evaluation Metrics

We report success rate as the primary metric for Panda-Gym and AntMaze, using the success indicators provided by the environment as the basis. For MiniGrid, we report mean episodic return, as

**Algorithm 2** Episode Priority Scoring

**function** SCOREEPISODE($e$)
  Let $g_e$ be the original environment goal of episode $e$
  **for** each valid timestep $\tau$ in $e$ **do**
    $c_\tau^e \leftarrow C(\rho(s_\tau, g_e))$
    Select the priority candidate

$$z_\tau^{\text{prio}} \leftarrow \arg\max_{z \in \mathcal{Z}} A(c_\tau^e, z)$$

    Compute

$$q_\tau \leftarrow u_\tau^H(z_\tau^{\text{prio}}; g_e)$$

  **end for**

$$p(e) \leftarrow \varepsilon_p + \frac{1}{|\mathcal{C}(e)|} \sum_{c \in \mathcal{C}(e)} \max_{\substack{\tau \in e \\ c_\tau^e = c}} q_\tau$$

  **return** $p(e)$
**end function**

shown in Figure 3. All learning curves show the mean across six random seeds, with shaded regions denoting one standard deviation.

For aggregate comparisons, we report four metrics. Final SR is the success rate at the final evaluation checkpoint. Best SR is the maximum success rate achieved during training. AUC is the area under the success-rate curve, normalized by the training budget so that larger values indicate better sample efficiency. And $t_{0.8}$ is the first timestep at which the mean success rate reaches $0.8$, if a method never reaches this threshold, we report -.

For the main AUC and $t_{0.8}$ comparisons, we compute $95\%$ bootstrap confidence intervals by resampling seeds with replacement. When comparing two methods on the same task, we use paired bootstrap tests over matched seeds.

## E ADDITIONAL MULTI-STEP EVIDENCE

Figure 7 provides a visual summary of the multi-step evidence ablation reported in Table 1. All variants use the same PickAndPlace Biased-Distractor environment, base learner, candidate set, and training budget; they differ only in the teacher update target used to estimate candidate utility.

The results show a clear separation between local evidence and window-based evidence. One-step progress and endpoint-only progress produce low AUC and never reach $0.8$ success. In contrast, both multi-step variants succeed, with the max-window aggregation achieving the highest final success, best success, AUC, and fastest $t_{0.8}$. This supports the design choice in Eq. 5: for manipulation tasks,

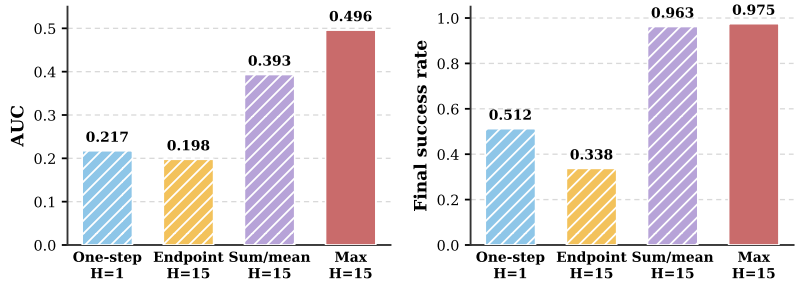



Figure 7: Multi-step evidence aggregation ablation on PickAndPlace Biased-Distractor. One-step and endpoint-only evidence fail to reach $0.8$ success, while window-based multi-step evidence succeeds. Max-window aggregation performs best across final success, AUC, and $t_{0.8}$.

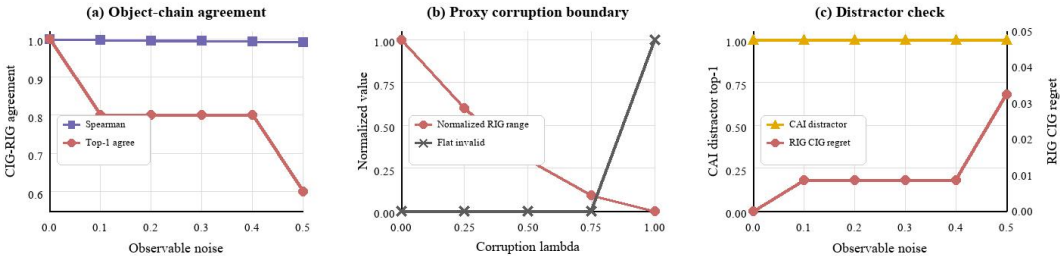



Figure 8: Additional exact tabular CIG-RIG diagnostics. (a) RIG ranking agreement with exact CIG under observable proxy noise. (b) Proxy-corruption boundary: as task information is removed, the RIG score range collapses and rankings become flat. (c) CAI-proxy ranks distractors first, while RIG maintains low CIG regret when the proxy is informative.

useful progress such as contact, grasp, or lift may occur within the future window but disappear by the endpoint, so taking the maximum over the window preserves transient evidence that endpoint-based scoring misses.

## F Additional CIG-RIG Diagnostics

We examine CIG–RIG ranking correspondence in a controlled tabular evidence model. Here, RIG denotes exact $\mathrm{RIG}_{\mathrm{MI}}$, not the potential-increment score used by Online-RIG. This diagnostic illustrates the role of proxy informativeness; it does not establish calibration of the operational score in continuous control.

**Experimental setup.** The latent variable $X \in \{0, \ldots, 5\}$ represents six semantic stages: far, near-contact, grasped, lifted, near-goal, and placed/release. These stages define a static evidence model, not a simulated transition process. We evaluate five anchor contexts $c \in \{0, \ldots, 4\}$, assigning probability $0.7$ to $X = c$ and distributing the remaining $0.3$ equally among its adjacent stages.

The candidate set contains six task candidates (approach, grasp, lift, move, place, and release), six corresponding distractor candidates, and one random candidate. Each candidate $z$ specifies a binary evidence variable $E_z \in \{0, 1\}$. For task candidates indexed by $k = 0, \ldots, 5$, the rows of the following matrix give $P(E_k = 1 \mid X = x)$ for $x = 0, \ldots, 5$:

$$P_{\mathrm{task}} = \begin{pmatrix} .88 & .25 & .08 & .08 & .08 & .08 \\ .35 & .88 & .25 & .08 & .08 & .08 \\ .28 & .35 & .88 & .25 & .08 & .08 \\ .28 & .28 & .35 & .88 & .25 & .08 \\ .28 & .28 & .28 & .35 & .88 & .25 \\ .28 & .28 & .28 & .28 & .35 & .08 \end{pmatrix}.$$

Distractor and random candidates have stage-independent evidence probabilities of $0.88$ and $0.50$, respectively. The observable proxy $Y$ takes values in the same six-state alphabet. We construct

$$p_c(x, y, e \mid z) = p_c(x)p(y \mid x)p(e \mid x, z),$$

and enumerate the joint distribution to compute $C_c(z) = I(X; E_z \mid c)$ and $R_c(z) = I(Y; E_z \mid c)$, using base-two logarithms.

**Proxy perturbations.** In the object-chain setting (Obj.), noise reverses the reported stage with probability $\epsilon$:

$$p_\epsilon(y \mid x) = (1 - \epsilon)\mathbf{1}[y = x] + \epsilon\mathbf{1}[y = 5 - x].$$

Thus, noise $0.1$ and $0.5$ correspond to reversal probabilities of $10\%$ and $50\%$, respectively. In the corruption setting (Corr.), the proxy is replaced by an independent uniform draw with probability $\lambda$:

$$p_\lambda(y \mid x) = (1 - \lambda)\mathbf{1}[y = x] + \lambda/6.$$

At $\lambda = 1$, the proxy contains no information about $X$.

Table 8: Exact tabular CIG-RIG diagnostics. Dist. and Rng. denote CAI distractor top-1 and RIG score range.

| Setting | Condition | Spearman | Top-1 | Regret | Extra |
|---|---|---|---|---|---|
| Obj. | noise 0.0 | 1.000 | 1.000 | 0.000 | Dist. 1.000 |
| Obj. | noise 0.1 | 0.997 | 0.800 | 0.009 | Dist. 1.000 |
| Obj. | noise 0.5 | 0.991 | 0.600 | 0.034 | Dist. 1.000 |
| Corr. | $\lambda = 0.0$ | 1.000 | 1.000 | 0.000 | Rng. 0.246 |
| Corr. | $\lambda = 0.5$ | 1.000 | 1.000 | 0.000 | Rng. 0.074 |
| Corr. | $\lambda = 1.0$ | – | – | – | Flat. 1.000 |

**Evaluation metrics.** All metrics are computed over the 13 candidates within each anchor and then averaged equally over the five anchors. Spearman measures rank correlation between $C_c$ and $R_c$, using average ranks for ties. Top-1 agreement records whether their maximizing candidate sets overlap. CIG regret is

$$\max_z C_c(z) - \max_{z \in \arg\max_{z'} R_c(z')} C_c(z),$$

using optimistic tie-breaking among RIG maximizers. The RIG score range is $\max_z R_c(z) - \min_z R_c(z)$; its plotted normalized value divides the anchor-averaged range by its value at $\lambda = 0$. An anchor is marked flat when its range is at most $10^{-12}$; Flat. reports the fraction of such anchors, for which ranking metrics are treated as undefined.

The task-agnostic comparator, denoted CAI-proxy, scores candidates by $A_c(z) = P(E_z = 1 \mid c)$. It is an evidence-probability proxy, not an implementation of the original CAI estimator. Dist. reports the fraction of anchors whose highest-scoring candidates are distractors.

Figure 8 provides additional diagnostics for the exact tabular experiments in Table 8. Since exact CIG can be enumerated in this controlled setting, we can directly compare the candidate ranking induced by operational RIG with the ideal CIG ranking.

Figure 8(a) shows the object-chain diagnostic under increasing observable proxy noise. Exact $\mathrm{RIG}_{\mathrm{MI}}$ maintains Spearman correlation above $0.99$ with CIG, while top-1 agreement decreases from $1.0$ to $0.6$. Thus, high overall ranking agreement does not guarantee preservation of the best candidate. These results support ranking correspondence in this controlled model, rather than universal calibration of the operational Online-RIG score.

Figure 8(b) shows the proxy-corruption boundary. As the observable proxy is mixed with independent uniform noise, the normalized RIG score range shrinks. At $\lambda = 1$, all candidate scores become zero up to numerical precision, and ranking metrics are undefined. This illustrates that information-based candidate discrimination requires an informative observable proxy.

Figure 8(c) compares RIG with CAI-proxy, the task-agnostic evidence-probability score defined above, rather than the original CAI estimator. CAI-proxy favors distractors by construction because their evidence occurs frequently but independently of task stage. In contrast, RIG incurs low CIG regret, distinguishing frequent evidence from evidence informative about latent task progress.

## G Additional Analysis of Context-Dependent Candidate Selection

The main text compares uniform selection and Online-RIG using the same unordered symbolic candidate set. This control isolates the teacher selection rule: both methods have access to the same dense progress signals, but only Online-RIG learns a context-dependent ranking over them. Therefore, the performance gap cannot be explained by the candidate set alone.

Uniform selection performs poorly because most candidate signals are useful only in particular progress contexts. For example, approach and contact signals are informative before interaction, grasp and lift become useful after contact, and move/place signals matter only once the object can be controlled. Sampling uniformly from the full unordered set frequently assigns reward to signals

that are physically premature or irrelevant to the current state. This produces noisy auxiliary rewards and weak curriculum guidance.

Online-RIG avoids this failure by maintaining a separate utility estimate for each context-candidate pair. The context-conditioned bandit can therefore assign high value to early-stage candidates near contact, shift toward grasp/lift after interaction, and later prefer move/place candidates near the goal. As reported in Table 2, this learned selection raises final success from $0.325$ to $0.988$ and reaches $0.8$ success at $0.84$M steps, while uniform selection never reaches the threshold. This supports the claim that Online-RIG is not merely exposing the learner to dense signals, it learns which signal is useful in each observable progress context.

# H Additional Component Ablations and Robustness

## H.1 Additional Component Ablations

Table 3 decomposes Online-RIG into two uses of the same progress evidence: curriculum reward and replay prioritization. The HER baseline uses the original sparse environment reward with uniform replay. RIG-P only keeps the reward unchanged but samples replay episodes according to stage-balanced RIG priority. RIG reward uses Online-RIG curriculum reward with uniform replay. RIG full combines both components.

The results show that RIG evidence is useful even when it does not directly modify the reward. RIG-P only improves final success from $0.325$ to $1.000$, demonstrating that RIG can identify episodes that are more useful for learning long-horizon progress. This rules out the interpretation that RIG is only a reward-shaping trick. Instead, the same progress evidence can also improve sample efficiency by changing the replay distribution.

The reward component is also strong. RIG reward reaches $0.988$ final success and has higher AUC than RIG-P only, indicating that dense curriculum reward provides more direct optimization signal during online interaction. The full method achieves the best AUC and the fastest $t_{0.8}$, showing that reward guidance and replay prioritization are complementary: the reward component shapes ongoing exploration and policy improvement, while RIG-P increases the probability of revisiting episodes that contain useful multi-stage progress evidence.

## H.2 Candidate-Source and Progress-Representation Robustness

Table 4 evaluates whether Online-RIG depends on a fully hand-designed symbolic candidate set. All variants use the same biased PickAndPlace-Distractor task and the same Online-RIG teacher, but differ in how candidate progress signals are represented.

Symbolic predicates provide the strongest compact prior. They encode manipulation-relevant events such as approach, contact, grasp, lift, move, and place, and therefore give the teacher a clean candidate vocabulary. This variant obtains the highest AUC, confirming that human-designed predicates remain a useful and sample-efficient representation.

However, Online-RIG remains effective with less manually specified feature candidates. Object-symmetric features match the symbolic variant in final and best success, despite including both task-object and distractor-object features. This indicates that the teacher can rank task-relevant candidates even when the candidate pool contains distractor-related alternatives. Achieved features also retain useful manipulation-progress information and reach high best success, although they are less sample efficient than symbolic predicates.

The achieved-goal xyz variant is substantially weaker. This is an important negative control: simply providing dense continuous targets is not sufficient. Candidate representations must preserve features related to manipulation progress, such as gripper-object distance, object height, object-goal distance, or task/distractor object identity. Thus, Online-RIG is robust to reduced manual design when the representation retains task-relevant structure, but is not expected to recover useful curricula from arbitrary or underspecified signals.

Additional landmark and random-feature controls further support this interpretation. Replay-derived landmark candidates and structured feature candidates can work well when they cover meaningful

progress regions, whereas unstructured random features are less reliable under distractors. These results suggest that RIG is candidate-source agnostic in the sense that candidates may come from symbolic predicates, replay features, landmark clusters, but the candidate source must expose observable task-progress information.

# I Additional Analysis of Task Relevance under Distractors

Figure 4 reports success rate curves on four PickAndPlace distractor layouts: *biased*, *matched*, *path*, and *random*. These layouts stress different failure modes: the distractor may be easier to contact, similarly reachable, located along the task path, or randomly placed. Across all four settings, RIG learns faster and reaches higher final success than DDPG+HER, CAI, and DISCOVER. The gap is largest in the biased and path layouts, where reachability-based or influence-based guidance can be attracted to distractor interactions rather than task progress.

To examine whether this improvement comes from task-relevant candidate selection rather than merely adding dense signals, Figure 5 reports a common candidate-ranking diagnostic. All methods are evaluated using the same candidate set and the same anchor states. Anchors are collected from scripted PickAndPlace trajectories and stratified into the observable progress contexts *far*, *near-contact*, *grasped*, *lifted*, *near-goal*, and *placed/released*. For each layout, identical anchors are used across methods.

The distractor top-1 rate is the fraction of anchor-context pairs for which the highest-scoring candidate is associated with the distractor object. Lower values indicate stronger task-relevant ranking. CAI frequently assigns the highest score to distractor candidates, reaching $0.907$ in the biased layout and $0.838$ in the matched layout. DISCOVER reduces this tendency but still ranks distractor-related candidates highest in many anchors. In contrast, RIG keeps the distractor top-1 rate low across all layouts: $0.111$, $0.102$, $0.095$, and $0.091$ for biased, matched, path, and random, respectively.

DDPG+HER does not contain a teacher or candidate-ranking module during training. Its value in Figure 5 is therefore a post-hoc diagnostic control rather than an additional HER component. Each candidate is scored from its compatibility with the subsequent HER replay trajectory on the same anchors, without context-conditioned RIG updates. This control tests whether ordinary HER replay experience alone induces a task-relevant ranking over the shared candidate set.

Overall, the diagnostic shows that RIG does not improve merely by rewarding novelty, reachability, or controllability. Its context-conditioned multi-step evidence preferentially ranks signals associated with long-horizon task progress, while suppressing distractor-related signals that are reachable or influential but irrelevant to the final goal.