SCALING CONTRASTIVE REINFORCEMENT LEARNING TO OPEN-ENDED ENVIRONMENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Developing generally capable agents requires scalable methods for learning a wide range of goal-directed behaviours. Contrastive reinforcement learning (CRL) offers a promising approach by reframing goal-conditioned value learning as self-supervised classification of an agent's own future states. We extend CRL to open-ended environments, where agents pursue semantic goals that represent desired outcomes rather than particular target states. We formalize these goals as predicates over the state space, which makes CRL's training labels computable in closed form, and derive a Rao-Blackwellized estimator of the resulting discounted goal-occupancy measure. We further find that this estimator is biased when an agent's actions can terminate the episode, as it implicitly conditions on the episode continuing and fails to penalize actions that cause termination. We address this bias with an absorbing class and termination-discounted targets that account for the probability mass beyond the episode end. The resulting critic serves as an effective teacher for a policy-gradient student. On Craftax, our method achieves comparable success to an agent trained with a per-goal Q-learning teacher across thousands of semantic goals while using up to $38\times$ less training compute. These results demonstrate that CRL can scale to large semantic goal spaces while accounting for irreversible failures in open-ended environments.

1 INTRODUCTION

Developing generally capable agents requires the ability to direct them towards a wide range of goals. Goal-conditioned reinforcement learning (GCRL; Kaelbling, 1993) is concerned with learning policies $\pi(a \mid s, g)$ that select actions a to reach a commanded goal g from state s, for instance by learning a goal-conditioned value function through temporal-difference (TD) learning (Schaul et al., 2015). Building on the self-supervised ideas behind breakthroughs in vision and language domains (Oord et al., 2019; Radford et al., 2021), contrastive reinforcement learning (CRL) has emerged as a promising alternative by reframing value learning as a classification problem (Eysenbach et al., 2023). CRL trains a classifier to predict future goal achievement by contrasting goals reached along the agent's trajectory with randomly sampled ones. Recent work has shown that CRL scales to very deep networks and continues to benefit from increasing model depth (Wang et al., 2026).

However, these results have largely been demonstrated in relatively restricted environments, such as gridworld navigation, robotic manipulation, and locomotion (Bortkiewicz et al., 2025). These settings simplify the goal-conditioned problem in three ways. First, goals correspond to particular target states, whereas in the real world goals are more semantic in nature and can correspond to arbitrary subsets of the state space. Second, each environment typically focuses on variations of the same underlying task, rather than a diverse set of qualitatively distinct objectives. Third, failure in these settings is usually limited to not reaching the commanded goal within the episode. In many environments, however, a single action can terminate the episode and with it the chance to reach any future goal. These simplifications break down as we move towards more open-ended environments.

In this work, we address these limitations by introducing three extensions to CRL, summarized in Figure 1. First, we define semantic goals as predicates over the state space, allowing each goal to be satisfied by multiple states and each state to satisfy multiple goals. This formulation makes the label distribution computable in closed form, from which we derive a Rao-Blackwellized estimator of the discounted goal-occupancy measure. Second, we show that CRL's targets are biased when the agent's actions can terminate the episode, as renormalizing over observed future states implicitly conditions on the episode continuing and fails to penalize actions that cause termination. We remove

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.

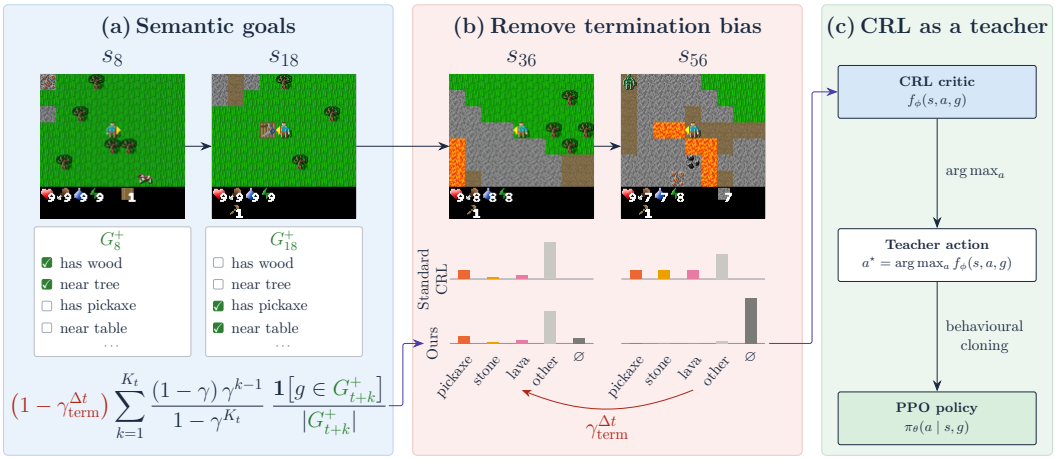


Figure 1: **Method overview.** We introduce three mechanisms to scale CRL to open-ended environments, illustrated on a Craftax episode of a trained agent. **(a)** We define goals as semantic predicates, so a single state can satisfy a set of goals G^+. Our Rao-Blackwellized target replaces the sampled future state with an exact expectation over all observed futures. Each future state contributes weight proportional to γ^{k-1}, split uniformly over the goals it satisfies. **(b)** Standard CRL has a termination bias because it renormalizes its targets over surviving futures. We instead assign the termination share $\gamma_{\text{term}}^{\Delta_t}$ of each target to an absorbing class $\varnothing$, which provides a learning signal for avoiding fatal actions. **(c)** We use the CRL critic trained on these targets to supervise a goal-conditioned PPO student via behavioural cloning.

this bias with an absorbing class that receives the probability mass beyond the episode end instead of redistributing it across the remaining goals. Termination-discounted targets then propagate this signal to preceding actions in the trajectory. Third, following the teacher-student design of Matthews et al. (2026), we use the improved CRL critic as a teacher for Proximal Policy Optimization (PPO; Schulman et al., 2017) and distil its all-goals value estimates into a single policy.

We evaluate our approach on Craftax (Matthews et al., 2024), a Minecraft-inspired benchmark in which an agent pursues goals in a procedurally generated environment. Craftax combines the challenges described above: its goals span qualitatively distinct objectives including resource acquisition, crafting, navigation, and fighting, while hazards such as lava and enemies can terminate the episode and prevent further goal achievement. Our results show that the proposed extensions improve CRL's learned critic, making it competitive with TD-trained value-based methods on Full Craftax. When scaled to thousands of goals, our CRL-based agent achieves comparable performance to an agent trained with a per-goal Q-learning teacher, requiring only 5% of the parameters and just over half the training time.

In summary, our contributions are:

1. We extend CRL to semantic goals and derive a Rao-Blackwellized estimator that makes full use of observed future states and eliminates variance from future-offset and label sampling.
2. We identify and correct a termination bias in CRL using an absorbing class and termination-discounted targets, which together lengthen survival and improve goal success.
3. We show that the improved CRL critic can distil its all-goals value estimates into a PPO policy, achieving performance comparable to that of a per-goal Q-learning teacher across thousands of goals at a fraction of the computational cost.

2 Preliminaries

Goal-conditioned reinforcement learning. We consider a goal-conditioned Markov decision process (MDP) $\langle \mathcal{S}, \mathcal{A}, \mathcal{G}, T, r, \gamma \rangle$ with state space $\mathcal{S}$, action space $\mathcal{A}$, goal space $\mathcal{G}$, transition kernel $T(s' \mid s, a)$, goal-conditioned reward function $r(s, g) = \mathbb{1}[s \in \mathcal{S}_g]$, and discount factor $\gamma \in (0, 1)$, where $\mathcal{S}_g \subseteq \mathcal{S}$ denotes the set of states that satisfy goal g (Andrychowicz et al., 2017). The agent

learns a goal-conditioned policy $\pi(a \mid s, g)$ that maximizes the expected discounted return with commanded goals drawn uniformly from $\mathcal{G}$. Value-based approaches learn a value function jointly over states and goals through TD learning (Schaul et al., 2015).

Discounted state-occupancy measure. The goal-conditioned value function can be expressed in terms of the distribution of states the agent visits in the future. For a policy π, the discounted state-occupancy measure of a state-action pair (s, a) is

$$\rho^\pi_\gamma(s' \mid s, a) = (1-\gamma) \sum_{k=1}^{\infty} \gamma^{k-1} \Pr^\pi(s_{t+k} = s' \mid s_t = s, a_t = a), \tag{1}$$

the distribution of the state visited k steps ahead when $k \sim \mathrm{Geom}(\gamma)$ (cf. Eysenbach et al., 2021; 2023). The goal-conditioned Q-function $Q^\pi(s, a, g) = \frac{1}{1-\gamma}\rho^\pi_\gamma(\mathcal{S}_g \mid s, a)$ is proportional to the probability that this future state satisfies the commanded goal.

Contrastive reinforcement learning. CRL estimates this probability directly and reframes learning a goal-conditioned value function as a classification task (Eysenbach et al., 2023). A state-action encoder $\phi(s, a)$ and a goal encoder $\psi(g)$ are trained so that the critic score $f(s, a, g) = \phi(s, a) \cdot \psi(g)$ corresponds to the future reachability of the goal g when taking action a from state s. During training, every state-action pair (s_t, a_t) acts as an anchor, the reference point against which candidate goals are scored. It is labelled with a future state of its own trajectory, which implicitly treats goals as target states with $\mathcal{S}_g = \{g\}$. The goal state $g^+ = s_{t+k}$ is selected by drawing an offset $k \sim \mathrm{Geom}(\gamma)$, so that positive samples are distributed according to $\rho^\pi_\gamma(\cdot \mid s_t, a_t)$. The encoders are trained using the InfoNCE loss (Oord et al., 2019), which identifies the positive goal g^+ among a set of negatives G^- sampled from other trajectories:

$$\mathcal{L} = -\mathbb{E}\left[\log \frac{\exp f(s_t, a_t, g^+)}{\exp f(s_t, a_t, g^+) + \sum_{g^- \in G^-} \exp f(s_t, a_t, g^-)}\right]. \tag{2}$$

In discrete action spaces, the policy $\pi(a \mid s, g)$ can be derived directly from the critic by acting ϵ-greedily on f. In continuous action spaces, an actor π_θ is trained to maximize the critic's score.

3 Methodology

We extend CRL with three mechanisms, which correspond to the panels of Figure 1. We first generalize the targets to semantic goals (Section 3.1), then correct them for episodes that end through the agent's own actions (Section 3.2), and finally distil the resulting critic into a policy-gradient student (Section 3.3).

3.1 CRL for Semantic Goals

We treat goals as predicates $P_g : \mathcal{S} \to \{0, 1\}$ with $P_g(s) = \mathbb{1}[s \in \mathcal{S}_g]$, indicating whether a state satisfies goal g. For a finite goal space $\mathcal{G}$, let $G^+(s) = \{g \in \mathcal{G} : P_g(s) = 1\}$ denote the set of goals a state satisfies. Classification is then performed over the full goal space rather than sampled negatives, using the categorical cross-entropy loss:

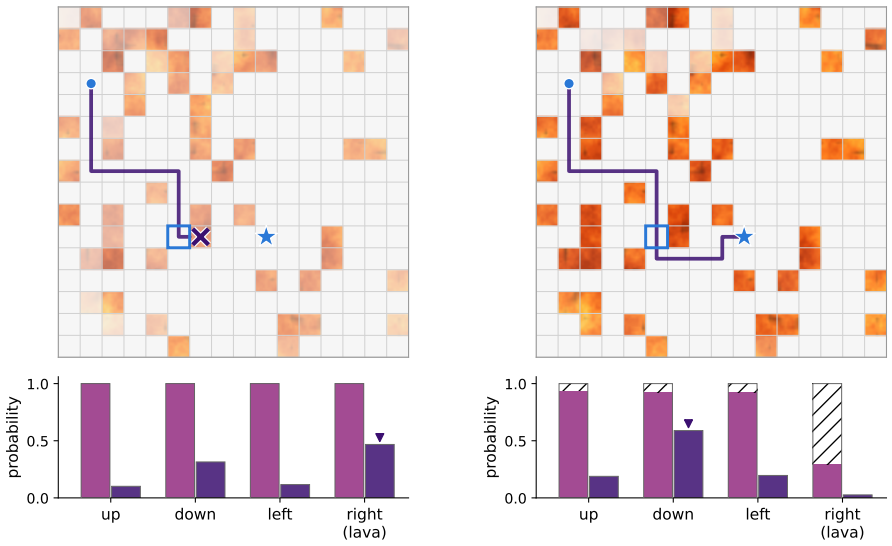
$$\mathcal{L} = -\mathbb{E}_{(s_t, a_t)}\left[\sum_{g \in \mathcal{G}} y_t(g) \log p(g \mid s_t, a_t)\right], \qquad p(g \mid s_t, a_t) = \frac{\exp f(s_t, a_t, g)}{\sum_{g' \in \mathcal{G}} \exp f(s_t, a_t, g')}, \tag{3}$$

where y_t is a target distribution and $p(g \mid s_t, a_t)$ is the softmax over the full goal space $\mathcal{G}$. Since a future state may satisfy multiple goals at once, the sampled label becomes a uniform soft target over the satisfied goal set:

$$y_t(g) = \frac{\mathbb{1}\big[g \in G^+(s_{t+k})\big]}{|G^+(s_{t+k})|}, \tag{4}$$

which is defined whenever the future state satisfies at least one goal. This target estimates the discounted goal-occupancy measure

$$\rho^\pi_\gamma(g \mid s, a) = (1-\gamma) \sum_{k=1}^{\infty} \gamma^{k-1}\, \mathbb{E}^\pi\left[\frac{\mathbb{1}\big[g \in G^+(s_{t+k})\big]}{|G^+(s_{t+k})|} \,\middle|\, s_t = s, a_t = a\right], \tag{5}$$



(a) Standard CRL (b) CRL + absorbing class

Figure 2: **Example of the termination bias in Gridworld Lava.** We train two critics on the same map and roll out the greedy policy of each from the start (•) to the commanded goal (★). Lava is drawn at the opacity the critic perceives it with, which is measured by its aversion to stepping in. At the boxed cell, the bars show per action the predicted mass on all goals (magenta) vs. on ∅ (hatched), and the preference for the commanded goal (blue, ▼ = greedy action). **(a)** Standard CRL renormalizes its targets over surviving futures and therefore predicts goal achievement for every action, including stepping into lava. It takes the fatal step *right* and dies (×). **(b)** The absorbing critic predicts termination (∅) for this action. It takes the safe step *down* and reaches the goal.

which extends the state-occupancy measure of Equation (1) from states to goals by splitting each state's probability mass uniformly over its satisfied goals. States that satisfy no goals contribute zero, and the missing mass is the probability of futures in which no goal is achieved. We return to this case in Section 3.2.

Because the label is computable in closed form for every future state, we can replace the offset sampling by its exact expectation as well:

$$y_t(g) = \sum_{k=1}^{K_t} \frac{(1-\gamma)\,\gamma^{k-1}}{1-\gamma^{K_t}} \frac{\mathbb{1}\left[g \in G^+(s_{t+k})\right]}{|G^+(s_{t+k})|}, \tag{6}$$

where K_t is the number of future steps available after t, limited by the window length W of the collected trajectories, by episode boundaries, and by an optional offset cap $k_{\max}$. The factor $1-\gamma^{K_t}$ renormalizes the truncated offset distribution.

Equation (6) is a Rao-Blackwellized estimator (Rao, 1945; Blackwell, 1947) of the discounted goal-occupancy measure. It replaces the random offset sampling with an average over all offsets in the trajectory, which preserves the mean and can only reduce the variance. While previous work could in principle also replace the offset sampling, it cannot replace the label sampling, because with state-based goals the label space is a continuum.

3.2 The Termination Bias

Equation (6) is a faithful estimate of the discounted goal-occupancy measure as long as episode ends are independent of the agent's behaviour (e.g. through time limits). However, if episodes can instead end through the agent's own actions, the truncation becomes informative, and the estimate biased. This problem is not unique to our semantic goal setting but applies more generally to CRL whenever future states are sampled from trajectories that the agent can terminate. To illustrate this issue and isolate it empirically, we introduce a modified version of the standard Gridworld environment where walls no longer block movement but end the episode when stepped into. We call this environment

Table 1: **CRL on the same 16×16 map with blocking walls, lava walls, and lava walls with the absorbing class added (mean $\pm$ std over three seeds).** All metrics are measured over training episodes at the end of 10^8 environment steps. Success is the fraction of commanded goals achieved, lava death the fraction of episodes ending in lava, and episode length the mean number of steps per episode. Turning the walls into lava collapses success and survival. Labelling fatal transitions with the absorbing class recovers both.

Environment	Method	Success rate	Lava death rate	Episode length
Gridworld	CRL	0.939 ± 0.004	—	175.0 ± 0.8
Gridworld Lava	CRL	0.635 ± 0.043	0.934 ± 0.013	31.1 ± 3.7
Gridworld Lava	CRL + absorbing class	0.935 ± 0.002	0.289 ± 0.010	146.8 ± 1.4

Gridworld Lava and visualize it in Figure 2. The agent is commanded a target cell and navigates a fixed wall layout to reach it. On success, a new random goal is commanded and the episode continues. In the standard environment, an action that would move the agent into a wall is a no-op, whereas in Gridworld Lava it terminates the episode.

Table 1 compares a CRL agent in the two environments. With lava walls, the success rate drops by 30 percentage points, and 93% of episodes end with the agent walking into lava. This failure originates in the future-state sampling that CRL methods share. For an anchor whose episode ends in Δ_t steps, γ^{Δ_t} of the offset distribution falls beyond the episode's end. Renormalizing over the surviving futures, as in Equation (6), redistributes this share over the goals achieved before death, thereby inflating each target by a factor of $1/(1-\gamma^{\Delta_t})$. The fatal transition itself has no future state and is excluded from the loss, while earlier anchors condition their targets on survival. No action therefore receives a learning signal reflecting the termination it caused. We refer to this problem as the termination bias and address it with two changes: an absorbing class that receives the unassigned target mass, and termination-discounted targets that extend the offset distribution beyond episode termination.

Absorbing class. We extend the classifier with an absorbing class, a $(|\mathcal{G}|+1)$-st candidate whose logit is fixed at zero:

$$p(g \mid s_t, a_t) = \begin{cases} \dfrac{\exp f(s_t, a_t, g)}{Z(s_t, a_t)} & \text{if } g \in \mathcal{G}, \\ \dfrac{1}{Z(s_t, a_t)} & \text{if } g = \varnothing, \end{cases} \qquad Z(s_t, a_t) = 1 + \sum_{g' \in \mathcal{G}} \exp f(s_t, a_t, g'). \tag{7}$$

The targets of Equation (6) remain unchanged, with the residual mass $y_t(\varnothing) = 1 - \sum_{g \in \mathcal{G}} y_t(g)$ assigned to the absorbing class. At the optimum, the classifier probabilities match the expected targets, and the absorbing class receives the residual mass. If episode ends are independent of the agent's behaviour, this recovers the goal-occupancy measure of Equation (5). A fatal transition, which previously had no valid label and was excluded from the loss, now becomes a positive example of the absorbing class. The last row of Table 1 shows that this change fully recovers the drop in success rate caused by the lava walls, reduces the lava death rate, and increases episode length.

Termination-discounted targets. The absorbing class penalizes the final action of an episode, but not the actions leading up to it. This is sufficient in environments such as Gridworld Lava, where a death is caused by a single fatal action. However, in long-horizon tasks, the responsibility for a death typically extends beyond the final step and is caused by either a single earlier mistake or multiple consecutive suboptimal actions. For the anchors preceding the final transition, the targets are still renormalized over the surviving timesteps, thereby assigning full target mass to the few goals achieved before death and overestimating the probability of reaching them. To fix this, we extend the offset distribution beyond episode termination:

$$y_t(g) = \left(1 - \gamma_{\text{term}}^{\Delta_t}\right) \sum_{k=1}^{K_t} \frac{(1-\gamma)\,\gamma^{k-1}}{1-\gamma^{K_t}} \, \frac{\mathbb{1}\left[g \in G^+(s_{t+k})\right]}{|G^+(s_{t+k})|}, \tag{8}$$

where Δ_t is the number of steps until the episode's observed end, with $\Delta_t = \infty$ where none is observed within the window. For each anchor, a portion $\gamma_{\text{term}}^{\Delta_t}$ of the target mass is redistributed

from the achieved goals to the absorbing class. The targets of all achieved goals are scaled down by the same factor, so actions whose futures end in death fall below safer alternatives for every commanded goal at once. The termination discount γ_{term} is decoupled from the sampling discount and controls how far back a death is penalized. Setting $\gamma_{\text{term}} = \gamma$ recovers the goal-occupancy measure of Equation (5) exactly, while smaller values restrict the penalty to the final steps before death.

3.3 The CRL Critic as a Teacher

The changes introduced so far enable the CRL critic to handle semantic goals and account for episodes terminated by the agent's own actions. The critic learns about the entire goal space, and the Rao-Blackwellized estimator allows it to learn from all future states observed along a trajectory. However, the critic only learns the goal-conditioned state-action scores $f(s, a, g)$ and does not directly optimize a parametrized policy. For discrete actions, CRL instead acts ϵ-greedily with respect to these scores. Policy-gradient methods such as PPO provide a complementary approach. They directly optimize a parametrized policy, but their reward signal is specific to the commanded goal. Matthews et al. (2026) combine both approaches in a method called Dual-LEO-PPO, where an all-goals Q-learner is trained alongside a PPO policy on the same experience and its greedy actions are distilled into the PPO policy using behavioural cloning. We replace the Q-learning teacher with the CRL critic, resulting in the following objective:

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{PPO}}(\theta) \; + \; \beta\, \mathbb{E}_{(s_t,\, g)}\Big[-\log \pi_\theta\big(a_t^* \mid s_t, g\big)\Big], \qquad a_t^* = \arg\max_a f(s_t, a, g), \tag{9}$$

where g is the commanded goal, $\mathcal{L}_{\text{PPO}}$ is the standard clipped PPO objective, and β controls the strength of the teacher's guidance and is annealed to zero over training. We call the resulting method Dual-CRL-PPO.

4 Experimental Evaluation

Our evaluation addresses three questions: how CRL-based agents compare against value-based and policy-gradient baselines, how the absorbing class and termination-discounted targets affect success and survival, and how CRL scales to thousands of goals.

4.1 Experimental Setup

We evaluate our approach on Craftax (Matthews et al., 2024), a Minecraft-inspired benchmark in which agents interact with procedurally generated worlds to accomplish goals involving navigation, resource acquisition, crafting, and combat. We adopt the goal-conditioned Craftax setup of Matthews et al. (2026) and consider two versions, illustrated in Figure 10: Craftax Classic, with 17 actions and 136 semantic goals, and Full Craftax, with 43 actions and 512 semantic goals. To study scaling to larger goal spaces, we additionally evaluate a setting in which goals are constructed from all possible values of the symbolic observation features, which gives 1,419 goals in Craftax Classic and 9,637 in Full Craftax.

We compare CRL and Dual-CRL-PPO against PPO (Schulman et al., 2017), Parallelised Q-Networks (PQN; Gallici et al., 2025), PQN with hindsight experience replay (HER; Andrychowicz et al., 2017), Learn Everything All at Once (LEO), Dual-LEO-PQN, and Dual-LEO-PPO (Matthews et al., 2026). Agents receive structured symbolic representations of the local map and agent state. All methods process observations with the same architecture, a convolutional backbone and MLP, and differ primarily in their goal interface. PPO and PQN receive the commanded goal as input, LEO uses a separate Q-value output for each goal, and CRL scores per-action state embeddings against learned goal embeddings. We build on PureJaxGCRL (Matthews, 2026), which provides JAX implementations of the baselines, and train every method for 10^9 environment steps. Baselines use the tuned settings of Matthews et al. (2026), and Table 8 gives an overview of all hyperparameters.

During training, commanded goals are sampled uniformly from the set of goals previously achieved by the agent. We report the greedy success rate, which is evaluated deterministically on every goal at the end of training. On the observation-level goal sets, greedy evaluation uses a stratified subset of 512 goals. Training curves show the training success rate, which is an exponential moving average of commanded-goal success of the behaviour policy during training. We run three seeds per configuration and report mean and standard deviation. We include extended experimental results in Appendix A and more details on our experimental setup in Appendix B.

Table 2: **Greedy success rate, training time, and parameter count on Craftax Classic (136 goals) and Full Craftax (512 goals), mean ± std over three seeds.** Time is in hours, Params in millions. We compare all baselines against CRL with and without the absorbing class and against Dual-CRL-PPO. The absorbing class leaves the success rate on Craftax Classic unchanged and slightly improves it on Full Craftax. Dual-CRL-PPO comes within two percentage points of Dual-LEO-PPO in both environments, with just over half the parameters on Full Craftax.

	Craftax Classic			Full Craftax		
	Success rate	Time	Params	Success rate	Time	Params
PPO	0.787 ± 0.019	1.6	8.75	0.123 ± 0.002	3.5	12.34
PQN	0.510 ± 0.030	2.9	4.58	0.098 ± 0.003	7.6	7.56
PQN+HER	0.735 ± 0.019	3.2	4.58	0.125 ± 0.006	9.5	7.56
LEO	0.650 ± 0.011	2.1	7.78	0.163 ± 0.012	5.7	30.35
Dual-LEO-PPO	$\mathbf{0.838 \pm 0.007}$	2.4	15.50	$\mathbf{0.268 \pm 0.011}$	6.4	41.06
Dual-LEO-PQN	0.733 ± 0.011	3.6	13.39	0.193 ± 0.003	9.6	39.55
CRL	0.564 ± 0.022	1.5	6.53	0.158 ± 0.012	3.9	10.64
CRL + absorbing	0.561 ± 0.022	1.5	6.53	0.168 ± 0.003	3.9	10.64
Dual-CRL-PPO	0.821 ± 0.004	2.2	15.29	0.255 ± 0.009	5.6	22.98

4.2 Success and Survival on Craftax

Each component of Section 3 addresses a shortcoming of standard CRL. On Craftax Classic, standard CRL trails all baselines except PQN (Table 2), struggling with longer-horizon goals such as acquiring coal and iron (Figure 5). On Full Craftax, it already outperforms PPO, PQN, and PQN+HER.

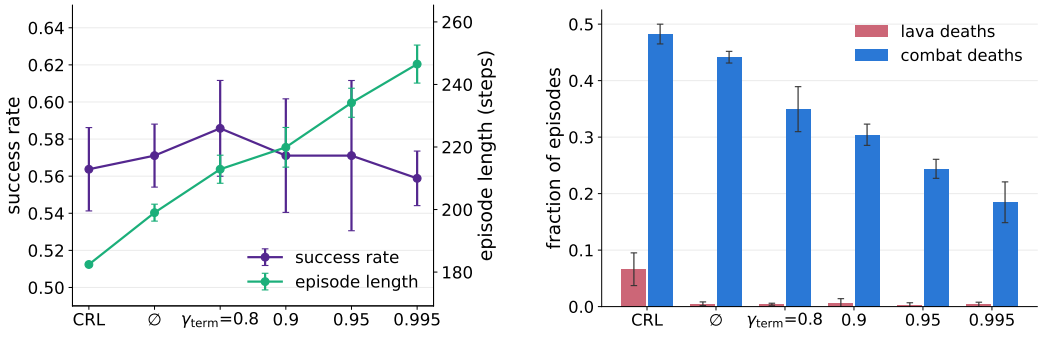
Termination bias. The absorbing class improves CRL's success rate on Full Craftax by one percentage point, outperforming every baseline except the Dual-LEO methods on average. On Craftax Classic, adding the absorbing class and Rao-Blackwellized targets eliminates lava deaths, which result from a single fatal action (Figure 3b), and increases episode length from 182 to 199 steps, while the success rate stays unchanged (Figure 3a). The termination discount additionally reduces combat deaths, which accumulate over multiple timesteps, from 44% to 18% of episodes and further increases episode length to 247 steps. At $\gamma_{\text{term}} = 0.8$, the added survival also translates into higher goal success, reaching 0.586. As γ_{term} approaches γ, survival continues to improve while goal success declines. The termination discount therefore trades goal success against survival by controlling how far the termination signal propagates back through the trajectory.

Sample efficiency. Agents trained with the Rao-Blackwellized estimator reach success thresholds with roughly 10 to 20% fewer environment steps than agents trained with sampled targets in otherwise identical configurations (Figure 7). The speedup persists in all five comparisons across varying termination discounts. The estimator also yields small but consistent gains in final success, with a higher mean in every case (Figure 8).

CRL as a teacher. Dual-CRL-PPO comes within two percentage points of Dual-LEO-PPO in both environments, with 15.3M versus 15.5M parameters on Craftax Classic and 23.0M versus 41.1M on Full Craftax. This difference in parameter counts follows from how the teachers represent goals. LEO maintains a separate output head per goal, so its parameter count grows with the number of goals and its goal set must be fixed before training. In contrast, CRL scores state embeddings against learned goal embeddings, making its parameter count nearly independent of the number of goals and allowing unseen goals to be scored with a forward pass through the goal encoder.

4.3 Scaling to Thousands of Goals

This difference in scaling becomes larger on the observation-level goal sets, which contain 1,419 goals on Craftax Classic and 9,637 on Full Craftax (Tables 5 and 6). At 9,637 goals, Dual-CRL-PPO uses 23.6M parameters compared with 443.2M for Dual-LEO-PPO, a $19\times$ reduction, and trains in 36 rather than 60 hours. Despite the smaller model, Dual-CRL-PPO remains within two percentage points of Dual-LEO-PPO in both settings, with success rates of 0.820 versus 0.840 on Craftax Classic and 0.171 versus 0.178 on Full Craftax. Measured by network-training FLOPs, Dual-CRL-PPO reaches its final success rate with $38\times$ less compute (Figure 4).



(a) Success rate and episode length

(b) Lava and combat death rates

Figure 3: **The absorbing class and termination-discounted targets extend survival on Craftax Classic (mean ± std over three seeds).** The x-axes show standard CRL, CRL with the absorbing class and Rao-Blackwellized targets (∅), and termination-discounted targets with increasing termination discount γ_{term} on top. Adding the absorbing class and Rao-Blackwellized targets eliminates lava deaths, which occur through a single fatal action, and lengthens episodes from 182 to 199 steps. The termination discount reduces combat deaths, which build up over multiple timesteps, from 44% of episodes to 18%, and lengthens episodes to 247 steps. The success rate is highest at $\gamma_{\text{term}} = 0.8$, reaching 0.586, and starts to decline as γ_{term} approaches γ.

5 Related Work

Goal-conditioned reinforcement learning. GCRL (Kaelbling, 1993) trains agents to achieve commanded goals. Standard RL algorithms such as PPO (Schulman et al., 2017) and PQN (Gallici et al., 2025) can be extended to this setting by conditioning their policies or value functions on the goal. Universal value function approximators (Schaul et al., 2015) learn a single value function jointly over states and goals. HER (Andrychowicz et al., 2017) improves sample efficiency by relabelling experience with goals achieved in hindsight. Horde (Sutton et al., 2011) and Q-map (Pardo et al., 2018) show how value estimates for many predictions or spatial goals can be learned in parallel from the same experience. LEO (Matthews et al., 2026) extends this idea to large discrete goal spaces, using per-goal value outputs to learn about all goals simultaneously. Its all-goals Q-learner can serve as a teacher for policy-gradient and value-based students in Dual-LEO-PPO and Dual-LEO-PQN. Dual-CRL-PPO retains this teacher-student design but replaces the Q-learning teacher with a contrastive critic whose parameter count is nearly independent of the number of goals.

Contrastive reinforcement learning. CRL (Eysenbach et al., 2023) builds on C-learning (Eysenbach et al., 2021) by reframing goal-conditioned value learning as classification of future states, avoiding TD backups on explicit rewards. CRL has been shown to scale to very deep networks (Wang et al., 2026) and to support emergent cooperation and exploration in multi-agent systems (Nimonkar et al., 2026). TD InfoNCE (Zheng et al., 2025) replaces the Monte Carlo estimate of discounted state occupancy with a bootstrapped temporal-difference estimate, enabling off-policy learning. Our Rao-Blackwellized estimator instead retains the Monte Carlo target while averaging exactly over future offsets and the semantic goals satisfied at each future state, reducing target variance without introducing bootstrap bias. Prior work has largely considered settings where goals correspond to target states, including navigation, manipulation, and locomotion. To the best of our knowledge, CRL has not been applied to open-ended, partially observable environments such as Craftax, where agents must survive while pursuing a wide range of semantic goals.

Termination bias in CRL. In concurrent work, Li et al. (2026) independently identify the termination bias, which they call catastrophic failure bootstrapping, and correct it with a mass-weighted InfoNCE objective and a learned survival term in the actor objective. The authors explicitly reject formulating failure as an absorbing outcome, as it requires special treatment in negative contrastive sampling and is not a valid commanded goal for the actor. Neither issue arises in our setting, since we compute the full softmax over the finite semantic goal set and the absorbing class is never commanded. In addition, our termination-discounted targets distribute the termination signal over the steps preceding a death, with γ_{term} setting the blame horizon.

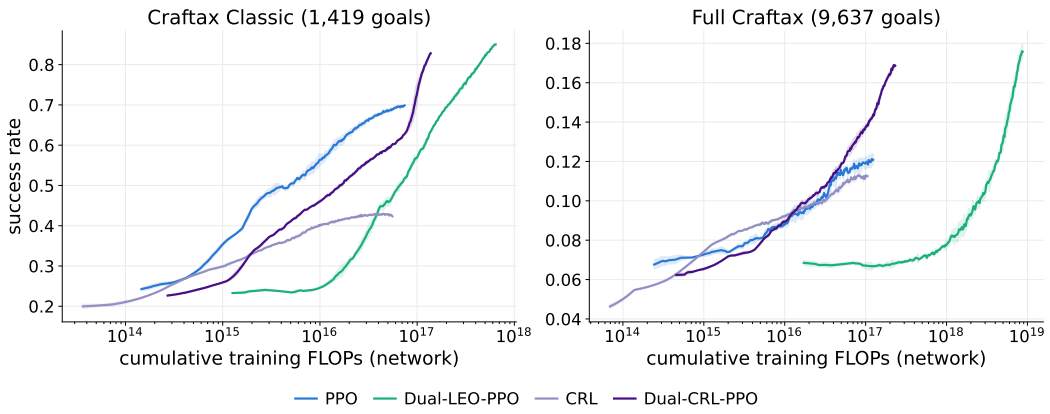


Figure 4: **Dual-CRL-PPO achieves comparable performance to Dual-LEO-PPO at a fraction of the training compute.** Training success rate (mean $\pm$ std over three seeds, shaded bands) as a function of cumulative training FLOPs on the observation-level goal sets. The count reflects the training compute of the networks only. On Full Craftax, Dual-CRL-PPO comes within one percentage point of Dual-LEO-PPO's final success rate with $38\times$ less compute. For clarity, we show PPO, Dual-LEO-PPO, CRL, and Dual-CRL-PPO. Figure 9 shows all baselines and the full measurement protocol.

6 Limitations and Future Work

Our approach has several limitations. First, we assume a well-defined goal set with known predicates that map states to the goals they satisfy. Second, both the co-satisfaction labels and the absorbing class rely on a full softmax over a finite goal set. Goal spaces that cannot be enumerated would require sampled negatives, and calibrating the absorbing class's share of the target mass under sampling remains an open problem. Third, the termination-discounted targets assign blame based on temporal proximity to termination. Every action within the blame horizon is downweighted, regardless of whether it contributed to termination. The termination discount γ_{term} therefore requires tuning to balance goal success and survival.

Addressing these limitations provides several directions for future work, including extending the method to learned or non-enumerable goal spaces and developing more targeted methods for propagating the termination signal to preceding actions. Other promising directions for extending CRL to open-ended environments include learning compositional goal representations, generalizing to unseen goals, and following natural-language goal descriptions. Allowing CRL agents to discover their own goals and subgoals could enable them to continually acquire new skills and generate their own curricula. Recent work has also shown that CRL can be successfully applied to multi-agent settings (Nimonkar et al., 2026). Extending this to environments such as Craftax-MA and Craftax-Coop (Omari et al., 2025) could enable CRL agents to pursue collective objectives in open-ended environments, including settings where agents have distinct roles and skills.

7 Conclusion

We extended contrastive RL to open-ended environments with large semantic goal spaces. We formulated semantic goals as predicates over the state space and derived a Rao-Blackwellized estimator of the resulting discounted goal-occupancy measure, which sped up learning. We further identified and addressed the bias introduced by episode termination through an absorbing class and termination-discounted targets, which together improved goal success and lengthened survival. Using a custom gridworld environment, we showed that this termination bias arises independently of semantic goals and that the absorbing class effectively reduces terminations caused by an agent's individual fatal actions. As a teacher for PPO, the improved CRL critic scales to thousands of semantic goals on Craftax and performs comparably to a per-goal Q-learning teacher while requiring substantially fewer parameters and less training compute. Together, our results are a step towards agents that can learn an increasingly broad range of goal-directed behaviours from their own experience.

AI USE STATEMENT

In this work, we used generative AI tools to assist with implementing the proposed methods, analysing and quantifying the termination bias, providing feedback on the experimental setup, and interpreting experimental results. Additionally, we used AI to write analysis and plotting code, identify and summarize related work, suggest the structure of sections, and draft and refine parts of the manuscript. We reviewed all AI-assisted work, including code, derivations, and numerical claims, and all drafted text was reviewed and edited by the authors. AI-assisted code was tested, and all tables and figures were generated programmatically from logged experimental data. We take responsibility for the final content of this work, including any text, claims, and outputs produced with the help of generative AI.

REFERENCES

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. Hindsight Experience Replay, July 2017. URL https://arxiv.org/abs/1707.01495v3. 2, 6, 8

David Blackwell. Conditional Expectation and Unbiased Sequential Estimation. *The Annals of Mathematical Statistics*, 18(1):105–110, 1947. ISSN 0003-4851. URL https://www.jstor.org/stable/2236107. 4

Michał Bortkiewicz, Władysław Pałucki, Vivek Myers, Tadeusz Dziarmaga, Tomasz Arczewski, Łukasz Kuciński, and Benjamin Eysenbach. Accelerating Goal-Conditioned RL Algorithms and Research, November 2025. URL http://arxiv.org/abs/2408.11052. arXiv:2408.11052 [cs.LG]. 1

Benjamin Eysenbach, Ruslan Salakhutdinov, and Sergey Levine. C-Learning: Learning to Achieve Goals via Recursive Classification, April 2021. URL http://arxiv.org/abs/2011.08909. arXiv:2011.08909 [cs.LG]. 3, 8

Benjamin Eysenbach, Tianjun Zhang, Ruslan Salakhutdinov, and Sergey Levine. Contrastive Learning as Goal-Conditioned Reinforcement Learning, February 2023. URL http://arxiv.org/abs/2206.07568. arXiv:2206.07568 [cs.LG]. 1, 3, 8

Matteo Gallici, Mattie Fellows, Benjamin Ellis, Bartomeu Pou, Ivan Masmitja, Jakob Foerster, and Mario Martin. Simplifying Deep Temporal Difference Learning. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 78148–78190, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/c23f3852601f6dd7f0b39223d031806f-Paper-Conference.pdf. 6, 8

Leslie Pack Kaelbling. Learning to Achieve Goals. In *International Joint Conference on Artificial Intelligence*, 1993. URL https://api.semanticscholar.org/CorpusID:5538688. 1, 8

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*, 2015. URL https://arxiv.org/abs/1412.6980. 17

Guopeng Li, Yiyang Duan, Yiru Jiao, and Chengcheng Xu. Arrive and Survive: Scaling Safe Goal-Conditioned Policy Learning from One-Bit Failure Signals, August 2026. URL https://arxiv.org/abs/2608.26571v1. 8

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the Variance of the Adaptive Learning Rate and Beyond, October 2021. URL http://arxiv.org/abs/1908.03265. arXiv:1908.03265 [cs.LG]. 17

Michael Matthews. MichaelTMatthews/purejaxgcrl, June 2026. URL https://github.com/MichaelTMatthews/purejaxgcrl. 6

Michael Matthews, Michael Beukman, Benjamin Ellis, Mikayel Samvelyan, Matthew Jackson, Samuel Coward, and Jakob Foerster. Craftax: A Lightning-Fast Benchmark for Open-Ended Reinforcement Learning, June 2024. URL http://arxiv.org/abs/2402.16801. arXiv:2402.16801 [cs.LG]. 2, 6, 17

Michael Matthews, Matthew Jackson, Michael Beukman, Thomas Foster, Alistair Letcher, Scott Fujimoto, Cédric Colas, and Jakob Foerster. Goal-Conditioned Agents that Learn Everything All at Once, May 2026. URL http://arxiv.org/abs/2605.23551. arXiv:2605.23551 [cs.LG]. 2, 6, 8, 17, 18, 19

Chirayu Nimonkar, Shlok Shah, Catherine Ji, and Benjamin Eysenbach. Self-Supervised Goal-Reaching Results in Multi-Agent Cooperation and Exploration, July 2026. URL http://arxiv.org/abs/2509.10656. arXiv:2509.10656 [cs.LG]. 8, 9

Bassel Al Omari, Michael Matthews, Alexander Rutherford, and Jakob Nicolaus Foerster. Multi-Agent Craftax: Benchmarking Open-Ended Multi-Agent Reinforcement Learning at the Hyperscale, November 2025. URL http://arxiv.org/abs/2511.04904. arXiv:2511.04904 [cs.LG]. 9

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation Learning with Contrastive Predictive Coding, January 2019. URL http://arxiv.org/abs/1807.03748. arXiv:1807.03748 [cs.LG]. 1, 3

Fabio Pardo, Vitaly Levdik, and Petar Kormushev. Scaling All-Goals Updates in Reinforcement Learning Using Convolutional Neural Networks, October 2018. URL https://arxiv.org/abs/1810.02927v2. 8

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision, February 2021. URL http://arxiv.org/abs/2103.00020. arXiv:2103.00020 [cs.CV]. 1

C. R. Rao. Information and the Accuracy Attainable in the Estimation of Statistical Parameters. *Bulletin of the Calcutta Mathematical Society*, 37:81–91, 1945. 4

Amit Sabne. XLA : Compiling Machine Learning for Peak Performance, 2020. 18

Tom Schaul, Daniel Horgan, Karol Gregor, and David Silver. Universal Value Function Approximators. In *Proceedings of the 32nd International Conference on Machine Learning*, pp. 1312–1320. PMLR, June 2015. URL https://proceedings.mlr.press/v37/schaul15.html. 1, 3, 8

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms, August 2017. URL http://arxiv.org/abs/1707.06347. arXiv:1707.06347 [cs.LG]. 2, 6, 8

Richard S. Sutton, Joseph Modayil, Michael Delp, Thomas Degris, Patrick M. Pilarski, Adam White, and Doina Precup. Horde: a scalable real-time architecture for learning knowledge from unsupervised sensorimotor interaction. In *The 10th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '11, pp. 761–768, Richland, SC, 2011. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 978-0-9826571-6-4. URL https://dl.acm.org/doi/10.5555/2031678.2031726. 8

Kevin Wang, Ishaan Javali, Michał Bortkiewicz, Tomasz Trzciński, and Benjamin Eysenbach. 1000 Layer Networks for Self-Supervised RL: Scaling Depth Can Enable New Goal-Reaching Capabilities, February 2026. URL http://arxiv.org/abs/2503.14858. arXiv:2503.14858 [cs.LG]. 1, 8

Chongyi Zheng, Ruslan Salakhutdinov, and Benjamin Eysenbach. Contrastive Difference Predictive Coding, October 2025. URL http://arxiv.org/abs/2310.20141. arXiv:2310.20141 [cs.LG]. 8

A Additional Results

A.1 Craftax Classic

Table 3: **Results on Craftax Classic (136 goals, mean ± std over three seeds).** Greedy is the success rate of the deterministic policy on every goal at the end of training, and Training the final value of an exponential moving average of commanded-goal success under the behaviour policy during training. ∅ denotes the absorbing class. The absorbing class leaves the success rate unchanged and extends survival (Figure 3). Rao-Blackwellized (RB) targets improve the mean success rate at every termination discount compared to sampled targets. Dual-CRL-PPO comes within two percentage points of Dual-LEO-PPO.

	Greedy	Training	Time (h)	Params (M)
PPO	0.787 ± 0.019	0.764 ± 0.012	1.6 ± 0.0	8.75
PQN	0.510 ± 0.030	0.497 ± 0.021	2.9 ± 0.0	4.58
PQN+HER	0.735 ± 0.019	0.726 ± 0.011	3.2 ± 0.0	4.58
LEO	0.650 ± 0.011	0.626 ± 0.008	2.1 ± 0.0	7.78
Dual-LEO-PPO	**0.838 ± 0.007**	**0.813 ± 0.003**	2.4 ± 0.0	15.50
Dual-LEO-PQN	0.733 ± 0.011	0.733 ± 0.011	3.6 ± 0.0	13.39
CRL	0.564 ± 0.022	0.550 ± 0.005	1.5 ± 0.1	6.53
CRL + ∅	0.561 ± 0.022	0.559 ± 0.008	1.5 ± 0.0	6.53
CRL + ∅ + γ_{term}=0.8	0.571 ± 0.024	0.563 ± 0.003	1.6 ± 0.0	6.53
CRL + ∅ + γ_{term}=0.9	0.569 ± 0.034	0.564 ± 0.010	1.5 ± 0.1	6.53
CRL + ∅ + γ_{term}=0.95	0.554 ± 0.011	0.566 ± 0.003	1.5 ± 0.1	6.53
CRL + ∅ + γ_{term}=0.995	0.537 ± 0.013	0.547 ± 0.005	1.6 ± 0.0	6.53
CRL + ∅ + RB	0.571 ± 0.017	0.566 ± 0.006	1.5 ± 0.0	6.53
CRL + ∅ + RB + γ_{term}=0.8	0.586 ± 0.026	0.577 ± 0.006	1.6 ± 0.1	6.53
CRL + ∅ + RB + γ_{term}=0.9	0.571 ± 0.031	0.572 ± 0.010	1.5 ± 0.0	6.53
CRL + ∅ + RB + γ_{term}=0.95	0.571 ± 0.040	0.566 ± 0.002	1.5 ± 0.0	6.53
CRL + ∅ + RB + γ_{term}=0.995	0.559 ± 0.015	0.556 ± 0.004	1.6 ± 0.1	6.53
Dual-CRL-PPO	0.821 ± 0.004	0.807 ± 0.006	2.2 ± 0.2	15.29

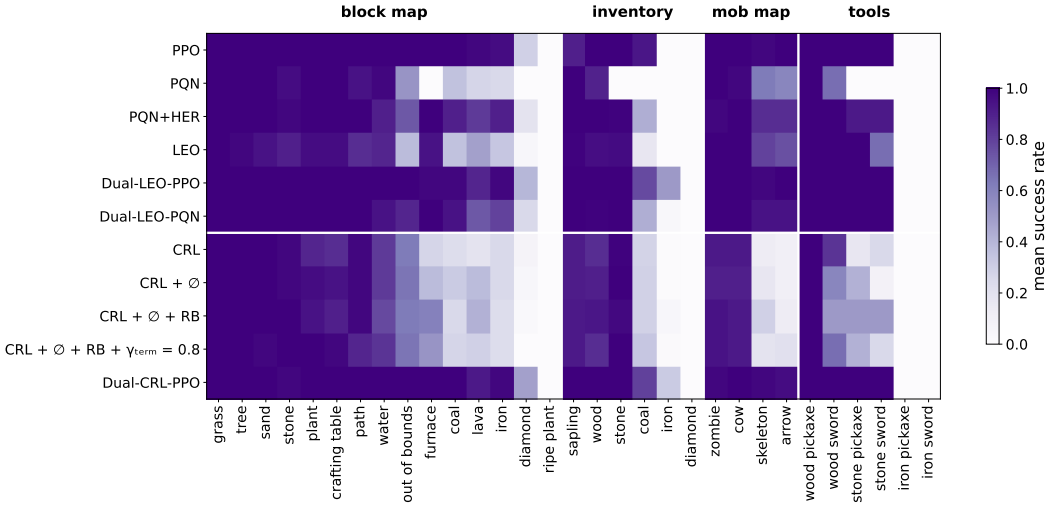


Figure 5: **Per-entity greedy success on Craftax Classic (mean over three seeds).** Each cell averages the greedy success of all goals involving that entity, over inventory counts, map directions, and seeds. ∅ denotes the absorbing class, and Table 3 lists all configurations. Columns are grouped by goal category and sorted from easy to hard within each group. CRL struggles with the harder goals, such as finding iron and crafting higher-tier tools, while Dual-CRL-PPO mirrors the strongest baselines.

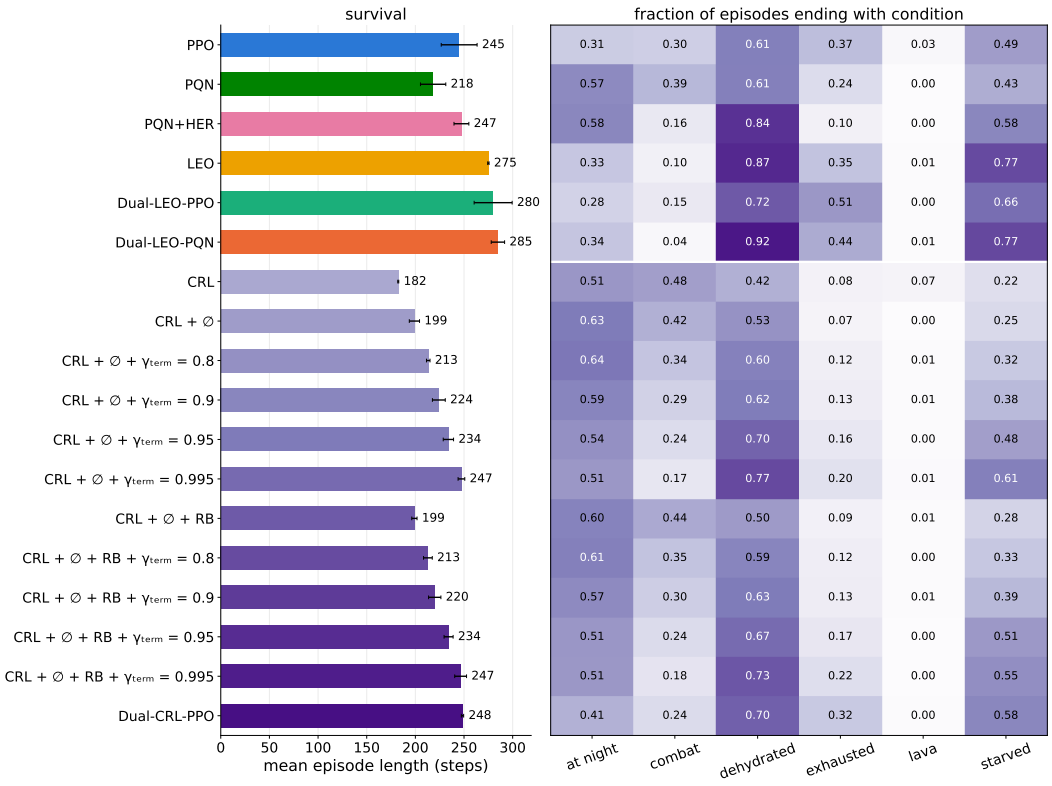


Figure 6: **Survival statistics on Craftax Classic (mean over three seeds).** Left: mean episode length, with error bars showing one standard deviation. Right: fraction of episodes that end while the given condition is active. Conditions can hold simultaneously, so rows need not sum to one. $\varnothing$ denotes the absorbing class. Standard CRL produces the shortest episodes, walks into lava most often, and dies in combat at the highest rate. Adding the absorbing class and termination-discounted targets lengthens survival, with larger γ_{term} moving episode lengths towards those of the baselines. The sampled and Rao-Blackwellized ladders show nearly identical survival statistics.

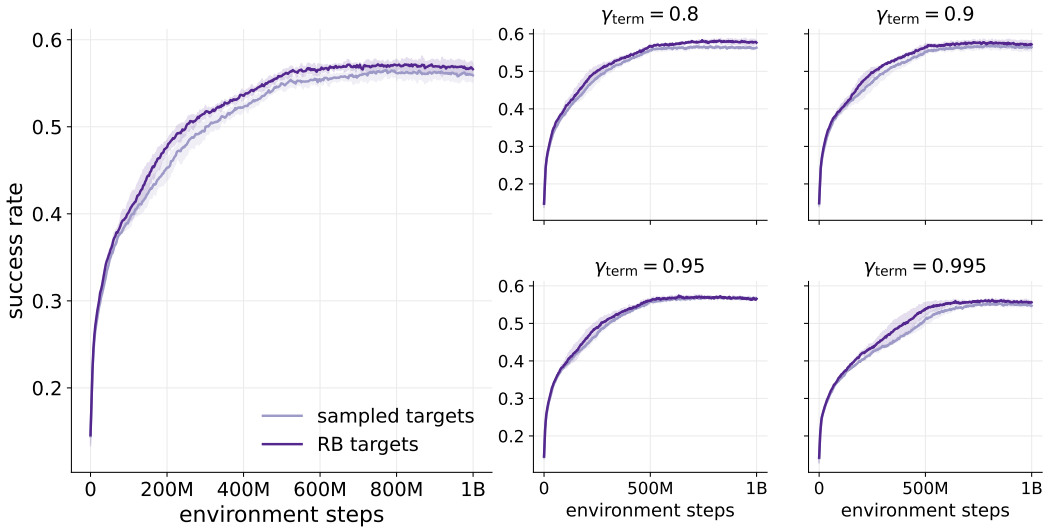


Figure 7: **CRL learns faster with Rao-Blackwellized targets.** Training success rate (mean $\pm$ std over three seeds, shaded bands) on Craftax Classic for CRL with the absorbing class, once with sampled targets and once with Rao-Blackwellized targets. Left: without termination-discounted targets. Right: the same comparison at each termination discount. Using the Rao-Blackwellized estimator speeds up learning and enables CRL to reach intermediate success thresholds with roughly 10 to 20% fewer environment steps. The speedup persists at every termination discount. Final success rates are similar, with the RB mean slightly higher in every configuration (Figure 8).

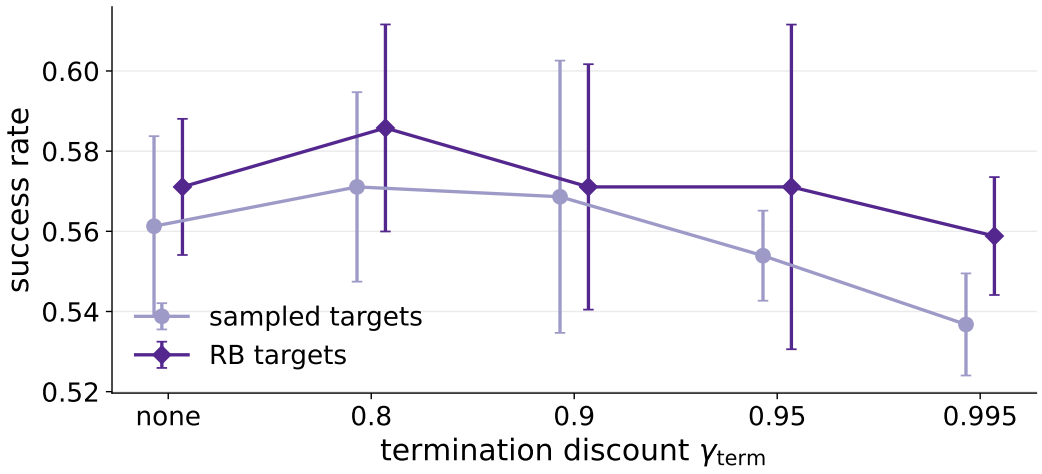


Figure 8: **Rao-Blackwellized targets improve the mean success rate at every termination discount (Craftax Classic, mean $\pm$ std over three seeds).** Greedy success of the sampled-target and Rao-Blackwellized (RB) ladders at matched termination discounts, with none denoting the absorbing class without termination-discounted targets. The RB mean is higher in all five matched comparisons, and the gap is largest at $\gamma_{\text{term}} = 0.995$.

A.2 Full Craftax

Table 4: **Results on Full Craftax (512 goals, mean ± std over three seeds).** Greedy is the success rate of the deterministic policy on every goal at the end of training, and Training the final value of an exponential moving average of commanded-goal success under the behaviour policy during training. The absorbing class slightly improves over standard CRL, and Dual-CRL-PPO comes within two percentage points of Dual-LEO-PPO with just over half the parameters.

	Greedy	Training	Time (h)	Params (M)
PPO	0.123 ± 0.002	0.121 ± 0.003	3.5 ± 0.1	12.34
PQN	0.098 ± 0.003	0.098 ± 0.001	7.6 ± 0.0	7.56
PQN+HER	0.125 ± 0.006	0.123 ± 0.001	9.5 ± 0.0	7.56
LEO	0.163 ± 0.012	0.162 ± 0.007	5.7 ± 0.2	30.35
Dual-LEO-PPO	**0.268 ± 0.011**	**0.263 ± 0.004**	6.4 ± 0.2	41.06
Dual-LEO-PQN	0.193 ± 0.003	0.183 ± 0.002	9.6 ± 0.2	39.55
CRL	0.158 ± 0.012	0.158 ± 0.003	3.9 ± 0.0	10.64
CRL + absorbing	0.168 ± 0.003	0.164 ± 0.003	3.9 ± 0.0	10.64
Dual-CRL-PPO	0.255 ± 0.009	0.252 ± 0.004	5.6 ± 0.3	22.98

A.3 Observation-Level Goal Sets

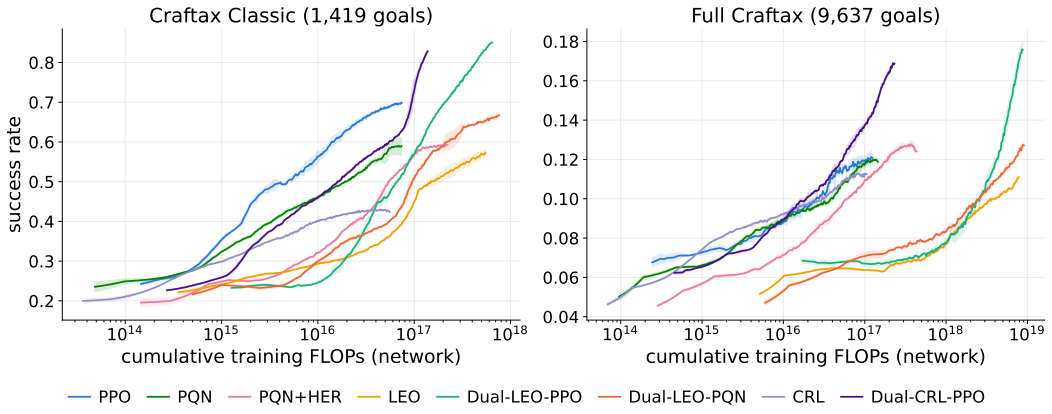


Figure 9: **Compute efficiency on the observation-level goal sets.** Training success rate (mean ± std over three seeds, shaded bands) as a function of cumulative training FLOPs. For each method, we measure the FLOPs of the forward and backward passes of all networks in one training step and multiply by the number of environment steps taken so far. Every method trains for 10^9 environment steps. Environment simulation, which is identical for all methods, and evaluation are excluded, so the count reflects the training compute of the networks only. On Full Craftax, Dual-CRL-PPO comes within one percentage point of Dual-LEO-PPO's final success rate with $38\times$ less compute.

Table 5: **Results on Craftax Classic with goals constructed from all values of the symbolic observation (1,419 goals, mean ± std over three seeds).** Greedy is the success rate of the deterministic policy at the end of training, evaluated on a stratified subset of 512 goals, and Training the final value of an exponential moving average of commanded-goal success under the behaviour policy during training. CRL and Dual-CRL-PPO use the absorbing class and Rao-Blackwellized targets with $k_{\max} = 32$, but no termination-discounted targets. Dual-CRL-PPO stays within two percentage points of Dual-LEO-PPO at less than half the parameter count.

	Greedy	Training	Time (h)	Params (M)
PPO	0.677 ± 0.012	0.700 ± 0.004	3.4 ± 0.0	8.75
PQN	0.583 ± 0.028	0.590 ± 0.027	6.3 ± 0.3	4.58
PQN+HER	0.591 ± 0.011	0.593 ± 0.002	7.0 ± 0.4	4.58
LEO	0.549 ± 0.022	0.573 ± 0.011	4.8 ± 0.2	30.14
Dual-LEO-PPO	$\mathbf{0.840 \pm 0.004}$	$\mathbf{0.851 \pm 0.001}$	6.1 ± 0.1	37.85
Dual-LEO-PQN	0.653 ± 0.016	0.667 ± 0.006	7.5 ± 0.1	35.75
CRL	0.419 ± 0.015	0.423 ± 0.001	2.7 ± 0.0	6.62
Dual-CRL-PPO	0.820 ± 0.008	0.828 ± 0.003	4.9 ± 0.1	15.37

Table 6: **Results on Full Craftax with goals constructed from all values of the symbolic observation (9,637 goals, mean ± std over three seeds).** Greedy is the success rate of the deterministic policy at the end of training, evaluated on a stratified subset of 512 goals, and Training the final value of an exponential moving average of commanded-goal success under the behaviour policy during training. CRL and Dual-CRL-PPO use the absorbing class and Rao-Blackwellized targets with $k_{\max} = 32$, but no termination-discounted targets. Dual-CRL-PPO achieves comparable performance to Dual-LEO-PPO with 5% of the parameters and just over half the training time.

	Greedy	Training	Time (h)	Params (M)
PPO	0.111 ± 0.010	0.121 ± 0.003	4.5 ± 0.1	12.34
PQN	0.122 ± 0.002	0.119 ± 0.002	5.4 ± 0.2	7.56
PQN+HER	0.117 ± 0.013	0.124 ± 0.001	36.0 ± 0.2	7.56
LEO	0.100 ± 0.006	0.111 ± 0.000	61.3 ± 0.3	432.53
Dual-LEO-PPO	$\mathbf{0.178 \pm 0.015}$	$\mathbf{0.176 \pm 0.004}$	60.4 ± 0.1	443.24
Dual-LEO-PQN	0.120 ± 0.004	0.127 ± 0.002	67.4 ± 0.2	441.74
CRL	0.112 ± 0.002	0.112 ± 0.000	32.9 ± 0.1	11.22
Dual-CRL-PPO	0.171 ± 0.008	0.169 ± 0.001	35.5 ± 0.2	23.56

B Experimental Details

B.1 Environments and Goal Sets

Environment. Craftax Classic and Full Craftax (Matthews et al., 2024) are procedurally generated survival environments visualized in Figure 10. The agent observes an egocentric window of the map together with its inventory and intrinsic stats. Since the window covers only a small part of the world, the environment is partially observable. We use the symbolic observations, which encode this information as structured features rather than pixels. All goal predicates are defined on the observation, so the agent can always verify whether a goal is achieved from its own view.



(a) Craftax Classic (b) Full Craftax

Figure 10: **Example scenes rendered as pixels.** (a) The Craftax Classic overworld. (b) A dungeon in Full Craftax, one of nine underground levels. Full Craftax has a larger field of view, more player statistics, and more items than Craftax Classic.

Semantic goal sets. Goals are binary masks over the symbolic observation, and the semantic goal sets follow Matthews et al. (2026). Table 7 lists the goal categories and their counts. Whether a goal is satisfied is read directly off the observation, and during training we check all goal predicates at every visited state to obtain the soft co-satisfaction labels of Equation (4).

Observation-level goal sets. For the scaling experiments, one goal is constructed per feature of the flattened symbolic observation, satisfied whenever that feature is active. On Craftax Classic, features that can never be active are removed, such as the player standing on a non-walkable block or a mob, zero health, and block types that never appear, which leaves 1,419 goals. On Full Craftax all features are kept, giving 9,637 goals.

B.2 Training and Evaluation

Training. All methods collect experience with 1,024 parallel environments and train for 10^9 environment steps. Commanded goals are sampled uniformly from the set of goals previously achieved by the agent. Each method alternates between collecting a rollout and updating its networks with minibatched gradient steps. Rollout lengths, epochs, minibatch sizes, learning rates, and discount factors are listed in Table 8. Gradients are clipped to a global norm of 1. The PPO policies are optimized with Adam (Kingma & Ba, 2015) and all other networks with RAdam (Liu et al., 2021).

Evaluation. At the end of training, we evaluate the deterministic policy on every goal, running four episodes per goal in parallel test environments and reporting the fraction of successful episodes. On the observation-level goal sets, this evaluation uses a fixed subset of 512 goals. The goals are allocated to categories proportionally to category size, with at least one goal per category, and sampled once with a fixed seed, so every run evaluates the same subset. The training success rate is an exponential moving average of commanded-goal success under the behaviour policy, with a decay of 0.999 per 2,048 environment steps, and the tables report its final value.

Compute. Every run uses a single NVIDIA H100 GPU. Per-run training times are listed in the result tables. The 129 reported runs total approximately 1,300 GPU hours.

Table 7: **Composition of the semantic goal sets.** The goal sets follow Matthews et al. (2026). Each goal is defined by one or more features of the symbolic observation and is achieved in every state whose observation activates at least one of them. Adjacent cells are the four cells directly next to the agent, one goal per entity type and direction. Tool goals select all counts of a tool, so they are achieved at any count.

Category	Example	Description	Classic	Full
Inventory	`wood_5`	Holding exactly n units of a resource	54	198
Tools	`stone_pickaxe`	Possessing a tool	6	17
Blocks	`TREE_left`	A block type in an adjacent cell, one goal per type and direction	60	132
Mobs	`zombie_up`	A mob type in an adjacent cell, one goal per type and direction	16	116
Placed items	`TORCH_down`	A placed item in an adjacent cell, one goal per type and direction		16
Dungeon level	`dlvl_3`	Being on a given dungeon floor		9
Enchantments	`helmet_fire`	An equipment piece carrying a given enchantment		12
Intrinsics	`intelligence_4`	A character attribute at a given level		12
Total			136	512

FLOPs measurement. We measure training compute with the cost analysis of the Accelerated Linear Algebra (XLA) compiler (Sabne, 2020), which counts the floating-point operations of one compiled training update, including the forward and backward passes of all networks. We divide by the environment steps consumed per update to obtain the FLOPs per environment step, and multiply by the number of environment steps taken so far to obtain the cumulative compute on the figures' x-axes. Environment simulation and evaluation are excluded, so the count reflects the training compute of the networks only.

B.3 Hyperparameters

Table 8: **Hyperparameters for all methods.** The hyperparameters for the baselines are the optimized settings from Matthews et al. (2026) and match the defaults of their implementation. The CRL column applies to all CRL ablations, which differ only in how the targets are constructed, as noted in the relevant sections. Entries of the form a / b for the dual methods give the setting of the policy and the teacher, respectively. Hyperparameters are identical across environments and goal sets, except that CRL uses $k_{\max} = 32$ on the observation-level goal sets. Dashes mark settings that are not applicable for a method.

	PPO	PQN	PQN+HER	LEO	Dual-LEO-PPO	Dual-LEO-PQN	CRL	Dual-CRL-PPO
total environment steps	10^9	10^9	10^9	10^9	10^9	10^9	10^9	10^9
parallel environments	1024	1024	1024	1024	1024	1024	1024	1024
rollout length	64	2	2	32	64	32	64	64
epochs per rollout	1	1	1	2	1 / 2	2	1	1
minibatch size	2048	256	2048	512	2048 / 512	512	2048	2048
learning rate	0.0002	0.0002	0.0001	0.0002	0.0002	0.0002	0.0002	0.0002
discount	0.995	0.995	0.995	0.99	0.995	0.99	0.995	0.995
max gradient norm	1	1	1	1	1	1	1	1
layer width	1024	1024	1024	1024	1024	1024	1024	1024
conv layers	1	1	1	1	1	1	1	1
conv features	32	16	16	32	32 / 16	32	32	32
conv kernel size	3	3	3	3	3	3	3	3
representation dim	—	—	—	—	—	—	64	64
HER mode	—	none	mixed	—	—	—	—	—
relabelled goals	—	—	2	—	—	—	—	—
BC coefficient	—	—	—	—	0.1	—	—	0.1
BC annealing	—	—	—	—	yes	—	—	yes
offset cap $k_{\max}$	—	—	—	—	—	—	—	32