Curiosity-Driven Exploration via Temporal Contrastive Learning

Faisal Mohamed[1] **Catherine Ji**[2] **Benjamin Eysenbach**[2,*] **Glen Berseth**[1,*]

faisal.mohamed@mila.quebec, cj7280@princeton.edu

[1]Mila-Quebec AI Institute, Université de Montréal [2]Princeton University

Abstract

Effective exploration in reinforcement learning requires not only tracking where an agent has been, but also understanding how the agent perceives and represents the world. To learn powerful representations, an agent should actively explore states that facilitate a richer understanding of its environment. Temporal representations can capture the information necessary to solve a wide range of potential tasks like goal reaching and skill learning while avoiding the computational cost associated with full-state reconstruction. In this paper, we propose an exploration method that leverages temporal contrastive representations to guide exploration, aiming to maximize state coverage as perceived through the lens of these learned representations. We demonstrate that such representations can enable the learning of complex exploratory behaviors in locomotion, manipulation, and embodied-AI tasks, revealing previously inaccessible capabilities and behaviors that traditionally required extrinsic rewards.

1 Introduction

Exploration remains a key challenge in reinforcement learning (RL), especially in tasks that demand reasoning over increasingly long horizons (Thrun, 1992) or with seemingly high-dimensional observations (Stadie et al., 2015; Burda et al., 2019b; Pathak et al., 2017). Perhaps the defining feature of RL, relative to other areas of ML, is the ability to find *new* strategies. Realizing this benefit would unlock important capabilities in robotics, LLM agents, and myriad other application domains. To unlock these capabilities, we need methods that can reduce the effective state space, but preserve the structure required for most tasks.

A common approach to exploration in reinforcement learning (RL) is to estimate the density of visited states. This estimated density is then used to construct an intrinsic reward function that encourages agents to visit novel states, thereby promoting state coverage. To enhance exploration in high-dimensional environments, prior work has proposed representation-based methods that learn compact representations, those sufficient to predict actions (Pathak et al., 2017), entirely random encodings (Burda et al., 2019b), or reconstructions of original observations (Stadie et al., 2015). These methods guide exploration along the *manifold* of meaningful states by discarding irrelevant information, as determined by the learned representations. However, identifying which aspects of the observation space are truly relevant remains a fundamental challenge. As a result, recent research has focused on learning representations that are aligned with the underlying task structure (Gelada et al., 2019; Zhang et al., 2021). In this paper, we ask: *How can we learn representations that facilitate exploration and are provably linked to the RL objective, retaining task-relevant features while excluding irrelevant ones?*

For a representation to be suitable for exploration, it should satisfy several key properties. Most importantly: (1) it should capture temporal relationships between current and future states, (2) it

*Equal Advising.

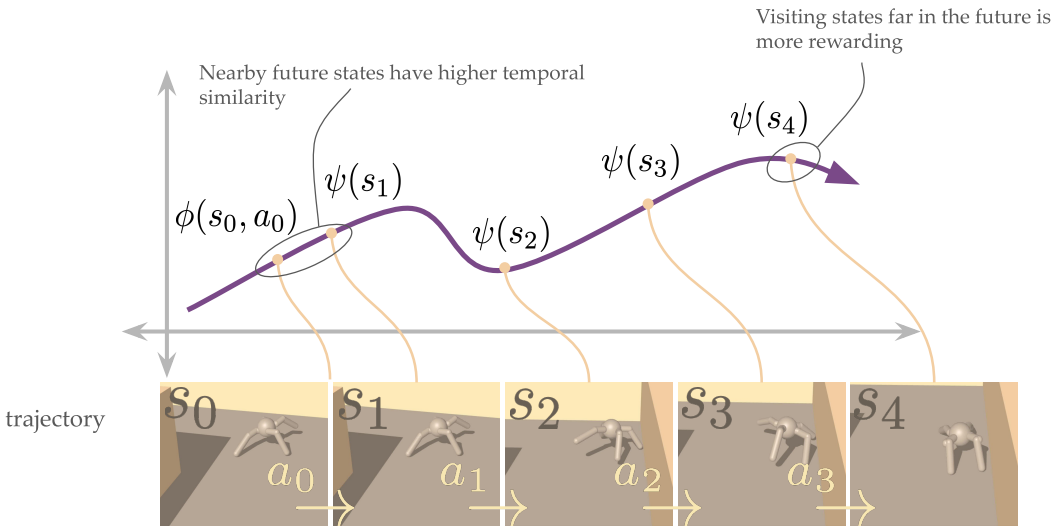

Figure 1: **Curiosity-Driven Exploration via Temporal Contrastive Learning** The agent's starting state is (s_0). We train a contrastive model such that the temporal similarity between the representation of (s_0, a_0) and $(s_{2,3,4,\ldots})$ should be high. We reward the agent for visiting states whose futures seem far away/improbable. For example, s_1 should confer less reward than the further s_4 to (s_0, a_0).

should scale with the dimensionality of the state space, and (3) it should remain up to date with the agent's ongoing experience.

One of the main challenges with previous works is the complexity of the representation learning process (Pathak et al., 2017; Burda et al., 2019b), limiting the agent's ability to keep the representation up-to-date with the agent's current distribution (Castanyer et al., 2024). Our proposed method aims to overcome these issues by building on temporal contrastive representations (Sermanet et al., 2018; Qian et al., 2021; Eysenbach et al., 2022; Dave et al., 2022), which are closely related to the successor representation (Dayan, 1993) and are provably sufficient to represent Q-values for any reward function (Mazoure et al., 2023). While prior work has primarily used these representations for encoding high-dimensional observations (Laskin et al., 2020) and for learning goal-directed skills (Eysenbach et al., 2022), we take a different direction: we use them to guide exploration. Specifically, we propose to reward the agent for visiting states with improbable futures from the perspective of representations, thereby encouraging it to explore less-visited regions of the environment.

The main contribution of this work is a new exploration algorithm that achieves state-of-the-art state coverage across navigation, manipulation, and open-world environments. Connections are made between this contrastive learning-based objective and information control objectives, where this objective prioritizes state-coverage through the lens of representations.

2 Related Work

Unsupervised RL. Prior work on unsupervised RL (Laskin et al., 2021) has proposed various task-agnostic methods for learning behaviors. A key direction in this area is intrinsic motivation, which encourages novelty-seeking behavior by maximizing state coverage or surprise. In low-dimensional and/or discrete environments, count-based exploration methods (Gardeux et al., 2016; Bellemare et al., 2016; Tang et al., 2017; Ostrovski et al., 2017; Martin et al., 2017; Xu et al., 2017; Machado et al., 2020) have demonstrated effective exploration in Atari games. However, these methods often struggle in high-dimensional or continuous state spaces. In such settings, prediction-error-based exploration approaches (Pathak et al., 2017; Burda et al., 2019a;b; Lee et al., 2019) have been more effective, both in video game environments and in continuous control tasks. Another line of research focuses on representation-based novelty, where a representation learning component is used to extract compact

features from raw state inputs. An entropy estimator is then applied to these learned representations to assess state novelty (Liu & Abbeel, 2021; Laskin et al., 2022).

A different approach to unsupervised RL involves training the agent to control the environment by either maximizing mutual information between states and actions (empowerment) (kly, 2005; Klyubin et al., 2005) or minimizing surprise(Friston, 2010; Berseth et al., 2021; Rhinehart et al., 2021). Empowerment-based methods (Biehl et al., 2015; Zhao et al., 2021; Mohamed & Jimenez Rezende, 2015; Karl et al., 2019; Hayashi & Takahashi, 2025; Levy et al., 2024; Jung et al., 2011; Du et al., 2020; Myers et al., 2024) encourage the agent to take actions that exert significant influence over future states, although solving the full problem remains intractable. In contrast, surprise minimization drives the agent to regulate the environment and maintain an orderly niche, giving rise to complex behaviors in both fully observed (Berseth et al., 2021; Hugessen et al., 2024) and partially observed settings (Rhinehart et al., 2021).

Representation learning for RL. Prior work on representation learning for RL focuses on self-supervised methods to improve the data efficiency of RL agents. A notable approach in this category involves the use of unsupervised auxiliary tasks, where a pseudo-reward is added to the task reward to shape the learned representations and provide an additional training signal. Examples of this approach include (Jaderberg et al., 2017; Farebrother et al., 2023; Oord et al., 2018; Laskin et al., 2020; Schwarzer et al., 2021). Another line of work focuses on forward-backward representations (Touati & Ollivier, 2021; Touati et al., 2023), which aim to capture the dynamics under all optimal policies and have been shown to exhibit zero-shot generalization capabilities. Moreover, contrastive learning has been applied in various exploration settings, including goal-conditioned learning (Eysenbach et al., 2022; Liu et al., 2025), skill discovery (Laskin et al., 2022; Yang et al., 2023; Zheng et al., 2025), and state coverage or curiosity (Liu & Abbeel, 2021; Du et al., 2021; Yarats et al., 2021). In the context of curiosity-driven exploration, (Du et al., 2021; Yarats et al., 2021) employ contrastive learning to learn visual representations in image-based environments, where the RL agent is trained to maximize the error of the representation learner (similar in spirit to prediction-error approaches). We consider C-TeC to fall under the state coverage category, while learning contrastive representations that facilitate density estimation and compress the agent experience into a low-dimensional space.

3 Background

We consider a controlled Markov process (i.e., an MDP without a reward function), defined by time-indexed states s_t and actions a_t. The initial state is sampled from $p_0(s_0)$, and subsequent states are sampled from the Markovian dynamics $p(s_{t+1} \mid s_t, a_t)$. Actions are selected by a stochastic, parameterized policy $\pi(a_t \mid s_t)$. Without loss of generality, we assume that episodes have an infinite horizon; the finite-horizon problem can be incorporated by augmenting the dynamics with an absorbing state. The key to C-TeC is to use a self-supervised, or intrinsic reward, built on temporal contrastive representations. We detail the necessary preliminaries below.

Discounted state occupancy measure Formally, we define the γ-discounted state occupancy measure conditioned on a state and an action (Ho & Ermon, 2016; Eysenbach et al., 2021; 2022) as

$$p(s_f \mid s, a) \triangleq (1-\gamma) \sum_{t=0}^{\infty} \gamma^t p(s_t = s_f \mid s, a), \tag{1}$$

where $p(s_t = s_f \mid s, a)$ is the probability of being at future state s_f at time step t conditioned on s, a. In continuous settings, the future state distribution $p(s_t = s_f \mid s, a)$ is a probability *density*.

Traditionally, the discounted state occupancy measure is defined with respect to a policy as $p_\pi(s_f \mid s, a)$. However, in this work, the intrinsic reward r_{intr} is defined using a discounted state occupancy measure over the trajectory buffer $\mathcal{T}$, which contains trajectories collected from a history of policies:

$$p_{\mathcal{T}}(s_f \mid s, a) \triangleq (1-\gamma) \sum_{0}^{\infty} \gamma^t p_{\mathcal{T}}(s_t = s_f \mid s, a).$$

To sample from the *trajectory buffer* distribution $p_{\mathcal{T}}(s_t = s_f \mid s, a)$, we first sample an offset $\Delta \sim \text{GEOM}(1-\gamma)$, then set the future state $s_f = s_{t+\Delta}$. Here, future state $s_f = s_{t+\Delta}$ is the state reached from (s, a) after executing Δ-number of actions within a sampled stored trajectory.

Contrastive learning Contrastive representation learning methods (cho, 2005; Oord et al., 2018; Chen et al., 2020) train a critic function C_θ that takes as input pairs of positive and negative examples, and learn representations so that positive pairs have similar representations and negative pairs have dissimilar representations. To estimate the discounted state occupancy, positive examples are sampled from a joint distribution $p_{\mathcal{T}}((s, a), s_f) = p_{\mathcal{T}}(s, a)p_{\mathcal{T}}(s_f \mid s_t, a_t)$, while the negative examples are sampled from the product of marginal distributions $p_{\mathcal{T}}(s, a)p_{\mathcal{T}}(s_f)$. Here, $p_{\mathcal{T}}(s_f)$ is the marginal discounted state occupancy:

$$p_\tau(s_f) = \int p_\tau(s_f \mid s, a)p_{\mathcal{T}}(s, a)\ \ ds\, da.$$

We use the InfoNCE loss to train the contrastive learning model (Oord et al., 2018). Let $\mathcal{B} = \{(s_t^{(i)}, a_t^{(i)}, s_f^{(i)})\}_{i=1}^K$ be the sampled batch, where $s_f^{(1)}$ is the positive example sampled from conditional distribution $p_{\mathcal{T}}(s_f \mid s_i, a_i)$ and $\{s_f^{(2:K)}\}$ are the $K-1$ negatives sampled from the marginal distribution $p_{\mathcal{T}}(s_f)$ (independently from (s_i, a_i)). In addition to the standard InfoNCE objective, prior work has shown that a LogSumExp regularizer is necessary for control (Eysenbach et al., 2021). The full contrastive reinforcement learning (CRL) loss is as follows:

$$\mathcal{L}_{\text{CRL}}(\theta) = -\mathbb{E}_{\substack{(s,a)\sim p_{\mathcal{T}}(s,a) \\ s_f^{(1)}\sim p_{\mathcal{T}}(s_f|s,a) \\ s_f^{(2:K)}\sim p_{\mathcal{T}}(s_f)}}\left[\log\left(\frac{e^{C_\theta((s,a),s_f^{(1)})/\tau}}{\sum_{j=1}^K e^{C_\theta((s,a),s_f^{(j)})/\tau}}\right) - 0.01\cdot\log\left(\sum_{j=1}^K e^{C_\theta((s,a),s_f^{(j)})/\tau}\right)^2\right]. \tag{2}$$

where τ is a temperature parameter. The optimal critic $C^*((s_t, a_t), s_f)$ corresponds to the following log probability ratio (Ma & Collins, 2018)

$$C^*((s_t, a_t), s_f) \approx \log \frac{p_{\mathcal{T}}(s_f \mid s_t, a_t)}{p_{\mathcal{T}}(s_f)},$$

where we use the following two parametrizations of the critic:

$$C_\theta((s_t, a_t), s_f)_{\ell_2} = -||\phi_\theta(s_t, a_t) - \psi_\theta(s_f)||_2 \tag{3}$$
$$C_\theta((s_t, a_t), s_f)_{\ell_1} = -||\phi_\theta(s_t, a_t) - \psi_\theta(s_f)||_1. \tag{4}$$

Conceptually, the critic C_θ gives a temporal similarity score between state-action pairs (s, a) and future states s_f via learned representation ϕ_θ and ψ_θ. A visual overview of the method is shown in Figure 1. These representations are powerful yet simple tools that capture complex temporal correlations between states-actions and future states. In our method C-TeC, we leverage these learned temporal contrastive representations to do exploration.

4 Curiosity-Driven Exploration via Temporal Contrastive Learning

To improve exploration, we learn representations that encode the agent's future state occupancy using temporal contrastive learning. We begin by describing how contrastive representation learning can be used to estimate state occupancy by learning a similarity function that assigns high scores to frequently visited future states and low scores to rarely visited ones (Eysenbach et al., 2022; Oord et al., 2018). We then explain how this similarity score can be leveraged to derive an intrinsic reward signal for exploration.

Algorithm 1 Curiosity-Driven Exploration via Temporal Contrastive Learning

Initialize: $\pi, \phi_\theta, \psi_\theta$,trajectory buffer $\mathcal{T}$
for each iteration **do**
  for each environment step $1 \leq t \leq T$ **do**
    $a_t \sim \pi(a_t \mid s_t)$
    $s_{t+1} \sim p(s_{t+1} \mid s_t, a_t)$
    $\tau_j \leftarrow \tau_j \cup \{s_t, a_t, s_{t+1}\}$
  $\mathcal{T} \leftarrow \mathcal{T} \bigcup \tau_j$
  Sample $\{(s_t^i, a_t^i)\}_{i=1}^{|\mathcal{B}|} \sim \mathcal{T}$ ▷ Sample a batch of state,action pairs
  Sample $\Delta_i \sim \text{GEOM}(1-\gamma)\ \forall i \in \{1, 2, \ldots, |\mathcal{B}|\}$ ▷ Sample a geometric offsets
  Set $s_f^i = s_{t+\Delta_i}^i \forall i \in \{1, 2, \ldots, |\mathcal{B}|\}$ ▷ Set the future state s_{f_i} according to Δ_i
  Compute intrinsic rewards: $\mathbf{r}_i = -C_\theta((s_t^i, a_t^i), s_f^i)$ ▷ Equation (5)
  Update representations: $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\text{InfoNCE}}(\mathcal{B} = \{(s_t^i, a_t^i, s_f^i)\}_{i=1}^{|\mathcal{B}|}; \theta)$ ▷ Equation (13)
  RL update using $\{(s_t^i, a_t^i, \mathbf{r}_t^i)\}_{i=1}^{|\mathcal{B}|}$ ▷ Update the policy using PPO/SAC

4.1 Training the contrastive model

As detailed in Section 3, the contrastive model $C_\theta(s_t, a_t, s_f)$ is trained on batches $\mathcal{B}$ of (s_t, a_t, s_f) tuples, where each s_f is sampled from the discounted future state distribution. Specifically, a geometric offset $\Delta \sim \text{GEOM}(1-\gamma)$ is sampled, and the future state is set to $s_f = s_{t+\Delta}$.

We use two parameterized encoders to define the contrastive model: $\phi_\theta(s_t, a_t)$ for state-action pairs and $\psi_\theta(s_f)$ for future states. A batch of state-action pairs $\{(s_t^{(i)}, a_t^{(i)})\}_{i=1}^K$ is passed through ϕ_θ, while the corresponding batch of future states $\{s_f^{(i)}\}_{i=1}^K$ is passed through ψ_θ. The resulting representations are then normalized to have unit norm. To compute the similarity between representations in practice, we found that using either the negative ℓ^1 or ℓ^2 norm was effective, depending on the environment. The contrastive encoder is trained to minimize the InfoNCE loss (Equation (13)) (Oord et al., 2018) . For each batch sample, the positive examples of other samples are treated as negatives, following common practice (Chen et al., 2020). The temperature parameter τ is learned during training as a learnable parameter. The details of the implementation are provided in Appendix A.

4.2 Extracting an exploration signal from the contrastive model

Given the contrastive model, a useful intrinsic reward can be constructed. Our aim is to reach unexpected but *meaningful* states. This is in contrast to surprise maximization or similar objectives which may prioritize unexpected but meaningless (i.e. random) states as those observed in the Noisy TV problem (see Figure 15) (Gruaz et al., 2024).

The contrastive model produces a similarity score between state-action pairs (s_t, a_t) and future states s_f proportional to the probability of reaching s_f from (s_t, a_t). *Negating* this similarity score results in our exploration signal r_{intr}, encouraging the agent to visit states that appear to have had improbable futures (in the eyes of the representations). Because the intrinsic reward has stochasticity from the future state sampling procedure (see Section 4.1), we write the expression for the expectation of r_{intr}:

$$\mathbb{E}[r_{\text{intr}}(s_t, a_t)] = \mathbb{E}_{p_\mathcal{T}(s_f \mid s_t, a_t)}\left[-C_\theta((s_t, a_t), s_f)\right] = \mathbb{E}_{p_\mathcal{T}(s_f \mid s_t, a_t)}\left[||\phi_\theta(s_t, a_t) - \psi_\theta(s_f)||\right] \quad (5)$$

where norm can be taken to be the ℓ^1 norm or ℓ^2 norm (See Section 6). This reward may seem counterintuitive – we should, perhaps, prioritize reaching states with high empowerment or surprise minimization. However, we claim that Equation (5) rewards meaningful exploration: the reward can also identify possible inconsistencies in the contrastive model, where the contrastive model assigns low likelihoods (large temporal distance) to actually encountered futures (see Section 5.2).

After learning the contrastive representations that define the r_{intr}, we train the parameterized policy $\pi(a_t \mid s_t)$ to maximize the discounted sum of rewards:

$$J(\pi) = \mathbb{E}_{\pi(a_t|s_t)}\left[\sum_{t=0}^{\infty} \gamma^t r_{\text{intr}}(s_t, a_t)\right]. \tag{6}$$

The experiments use PPO (Schulman et al., 2017) and SAC (Haarnoja et al., 2018b;a;c) for policy training (pseudocode in Algorithm 1). In practice, we found that using a single sample future state to approximate the expectation in Equation (5) works well, except in Craftax-Classic, where we used a Monte Carlo estimate. Additional details are provided in Appendix A.4.1.

4.3 Future states sampling

An important design choice is how to sample the future state when computing the intrinsic reward. For example, consistent sampling from states far in the future could lead to a high variance reward signal that might hinder the agent's learning, and sampling from the nearby states can be inefficient as the agent might already have a strong model over these states. One natural strategy is to sample according to the discounted occupancy measure, as described in Equation (1). Alternatively, we can sample uniformly from the future states (conditioned on the current state and action). We observe that sampling from the according to the discounted occupancy measure yields good performance across environments and we stick to this strategy in our experiments. We also show the performance differences between these sampling strategies in the experiments Section 6.3.

5 Interpretation of C-TeC

In the below sections, we provide intuition for how the representation-parameterized intrinsic reward may drive effective exploration behavior. Sec. 5.1 details an information-theoretic interpretation of C-TeC, ignoring learned representations to help build intuition and compare with other info-theoretic objectives (see Appendix E). In Section 5.2, we highlight the importance of the contrastive representations to C-TeC performance. Notably, contrastive representations enable C-TeC's performance to remain the same with or without noise in the Noisy TV environment (Figure 15).

5.1 Information-Theoretic Expression of C-TeC

The intrinsic reward has a corresponding information-theoretic interpretation. We consider the limit where representations perfectly capture the underlying point-wise MI. In this regime, the intrinsic reward evaluates to the negative of the KL-divergence between the conditional future-state distribution $p_\mathcal{T}(s_f \mid s_t, a_t)$ and the marginal future-state distribution $p_\mathcal{T}(s_f)$:

$$\begin{aligned}\mathbb{E}[r_{\text{intr}}(s_t, a_t)] &= -\mathbb{E}_{p_\mathcal{T}(s_f|s_t,a_t)}\left[\log \frac{p_\mathcal{T}(s_f \mid s_t, a_t)}{p_\mathcal{T}(s_f)}\right] \\ &= -D_{\text{KL}}[p_\mathcal{T}(s_f \mid s_t, a_t) \,||\, p_\mathcal{T}(s_f)] \leq 0. \qquad (D_{\text{KL}} \text{ is always non-negative.})\end{aligned} \tag{7}$$

This intrinsic reward describes mode-seeking behavior: notably, the conditional should only have support where the marginal $p_\mathcal{T}(s_f)$ has support. This optimization is distinct from minimizing the *forward* KL-divergence $D_{\text{KL}}[p_\mathcal{T}(s_f) \,||\, p_\mathcal{T}(s_f \mid s_t, a_t)] \geq 0$, which instead prioritizes mean-seeking behavior over regions of the state space where the marginal may *not* have support.

This mode-seeking behavior can be interpreted as prioritizing (s, a) with trajectories (futures) that look improbable or temporally-distant, but have been successfully achieved at some point:

$$\begin{aligned}\mathbb{E}[r_{\text{intr}}(s_t, a_t)] &= -D_{\text{KL}}[p_\mathcal{T}(s_f \mid s_t, a_t) \,||\, p_\mathcal{T}(s_f)] \\ &= \underbrace{H[S_f \mid s_t, a_t]}_{\text{surprise}} + \underbrace{\mathbb{E}_{p_\mathcal{T}(s_f|s_t,a_t)}[\log p_\mathcal{T}(s_f)]}_{\text{“familiarity” term}},\end{aligned}$$

where S_f denotes the future state *random variable* and $s_f \sim p_\mathcal{T}(s_f \mid s_t, a_t)$. In this form, we see that the intrinsic reward prioritizes spread-out trajectories ("surprise") over states that *have actually been seen* ("familiarity"). States encountered during roll-out are then added to the marginal, and the process repeats.

To test the hypothesis that this mode-seeking behavior is important, we ran experiments where the intrinsic reward is the *forward*, mean-seeking KL (**??**). Appendix C.5 shows that the objective succeeds because it is minimizing this mode-seeking formulation of the KL rather than fitting the conditional future states to a broad marginal. A linear stability analysis on the fixed points of C-TeC is in Appendix F.1; we simplify the problem setting for analysis. Notably, there are no easily-achievable stable fixed points for general nontrivial MDPs.

5.2 Representations are Necessary for C-TeC to Succeed

Temporal contrastive representations are crucial for effective exploration in C-TeC. Experimental results in Appendix C.4 show that the method is *not* robust to the usage of a monolithic critic $f(s, a, g)$, suggesting that the parameterization of the critic with representations is necessary to provide useful exploration signal.

Importantly, the representations not only capture a raw info-theoretic exploration signal but also a form of prediction error. All of the analysis in Section 5.1 assumes a fully-expressive critic that perfectly captures the point-wise MI. However, **the true learned representations only approximate the point-wise MI**. The full expected intrinsic reward is as follows:

$$\mathbb{E}[r_{\text{intr},\phi,\psi}(s_t, a_t)] = -\mathbb{E}_{p_\mathcal{T}(s_f \mid s_t, a_t)}\left[\log \frac{p_{\phi,\psi}(s_f \mid s_t, a_t)}{p_{\phi,\psi}(s_f)}\right]$$

where $p_{\phi,\psi}$ describe the (relative) probability distributions under the learned contrastive representations ϕ and ψ. Thus, the reward prioritizes exploration in areas with highly inefficient representation encoding schemes of future states – the future states look improbable to the representations.

Finally, representation learning helps enable the agent to ignore spurious noise like in the Noisy TV (see Fig. 15 results). The learned contrastive representations keep track of bits that distinguish future from random states. Noise randomly sampled from the same distribution every timestep *does not lead to stronger classifier performance*. Thus, the distance between representations and corresponding intrinsic reward and policy should, in principle, be independent of this noise.

6 Experiments

Our experiments show that contrastive representations can be used to reward the agent for visiting less-occupied or distant future states. We then use the C-TeC reward function for exploration in robotic environments and Craftax-Classic. We mainly study the following questions: (**Q1**) How well does C-TeC reward capture the agent's future state distribution? (**Q2**) How effectively does C-TeC explore in locomotion, manipulation, and Craftax environments compared to prior work? (**Q3**) How sensitive is C-TeC to the future state sampling strategy?

Environments We use environments from the JaxGCRL codebase (Bortkiewicz et al., 2025). Specifically, we evaluate C-TeC on the `ant_large_maze`, `humanoid_u_maze`, and `arm_binpick_hard` environments, which require solving long directed plans to reach goal states. In the maze-based environments, the agent's objective is to reach a designated goal specified at the start of each episode. Exploration in these settings corresponds to maze coverage: an agent that visits more unique positions in the maze demonstrates better exploration capabilities. In the `arm_binpick_hard` environment, which differs from the more navigation-themed tasks used in prior work, the agent must pick up a cube from a blue bin and place it at a specified target location in a red bin. This represents a challenging exploration task, as the agent must locate the cube, grasp it, and successfully place it at the correct target location.

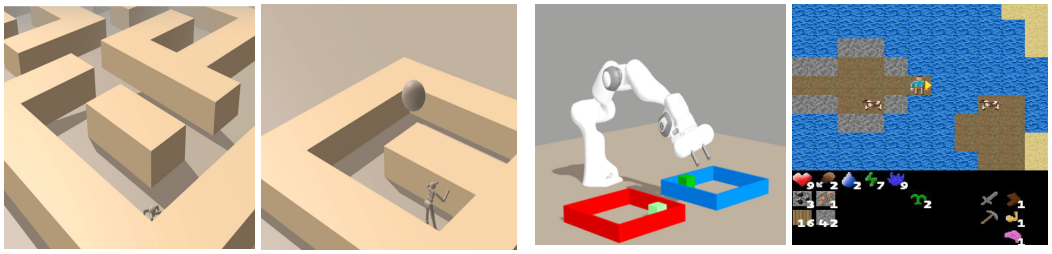

Figure 2: **Environments**. Maze coverage, robotic manipulation, and the survival game Craftax.

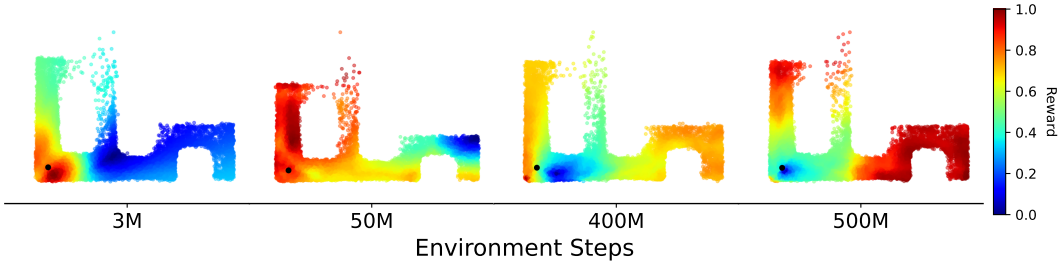

Figure 3: **Evolution of the C-TeC reward during training**. This figure shows how the intrinsic reward changes over the course of training based on future state visitation. The black circle in the lower-left corner represents the starting state. Early in training (3M steps), higher rewards are assigned to nearby states. As training progresses, the agent explores farther, and the reward increases for more distant regions. All reward values are normalized for visualization.

Our experiments with the `ant` and `humanoid` agents assess the method's ability to achieve broad state coverage using two complex embodiments. Meanwhile, the `arm_binpick_hard` task evaluates the method's effectiveness at exploration in an object manipulation setting. We also run C-TeC on Craftax-Classic (Matthews et al., 2024), a challenging open-world survival game resembling a 2D Minecraft. The agent's goal is to survive by crafting tools, maintaining food and shelter, and defeating enemies

In the locomotion and manipulation environments, we compare C-TeC to common prior methods for exploration: **Random Network Distillation (RND)** (Burda et al., 2019b) and **Intrinsic Curiosity Module (ICM)** (Pathak et al., 2017) which are both popular intrinsic motivation methods for exploration. **Active Pre-training (APT)** (Liu & Abbeel, 2021): APT learns observation representations using contrastive learning, where positives are augmentations of the same observation and negatives are different observations. It uses the KNN distance between state representations as an exploration signal, which correlates with state entropy. Unlike C-TeC, APT does not learn representations predictive of the future. In Craftax, we compare against RND, ICM, and **exploration via elliptical episodic bonuses (E3B)** (Henaff et al., 2022), a count-based exploration method. We found that using the negative L_1 distance (Equation (4)) as the critic function works best in the robotics environments, while the negative L_2 distance (Equation (3)) performs best in Craftax. A comparison of different critic functions can be found in the appendix.

6.1 Capturing the future state distribution (Q1)

The goal of this experiment is to demonstrate that the C-TeC reward captures the future state distribution. As a result, it can be used to incentivize the agent to visit less-occupied and more distant future states. We visualize the C-TeC reward at different stages of training in the `ant_hardest_maze` environment. The contrastive critic is defined as the negative L_1 distance (Equation (4)), and the policy is trained to maximize the intrinsic reward defined in Equation (5). Figure 3 shows the reward values in a section of the maze, with the black circle in the lower-left corner indicating the starting state. In the early stages of training (3M steps), the reward is highest for nearby states. As training progresses, the agent explores farther, and the reward increases for more distant regions (e.g., at 400M and 500M steps). Over time, the reward becomes increasingly aligned with the maze's geometry.

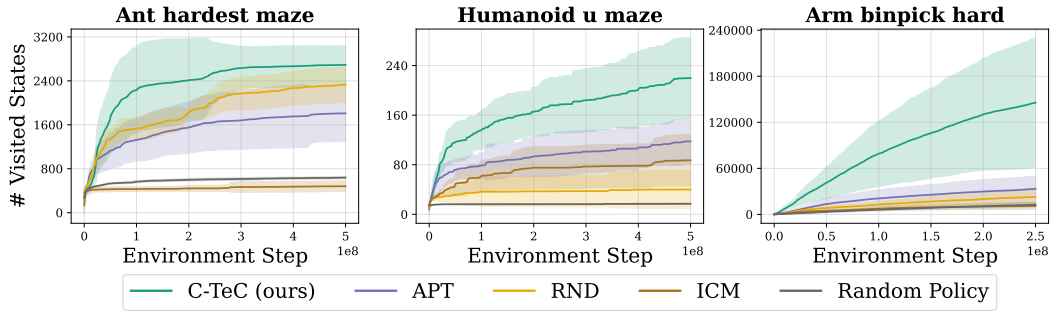

Figure 4: **C-TeC explores more states than prior methods.** We compare the state coverage of C-TeC to APT (Liu & Abbeel, 2021), RND (Burda et al., 2019b) and ICM (Pathak et al., 2017). We include a uniform random policy as well.

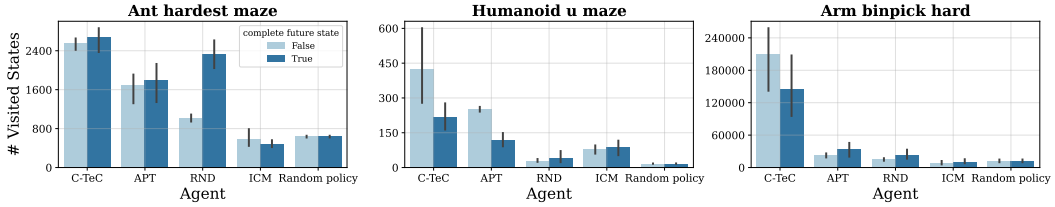

Figure 5: **State coverage when leveraging prior knowledge** C-TeC outperforms prior methods (Liu & Abbeel, 2021; Burda et al., 2018; Pathak et al., 2019) and can explore effectively when leveraging prior knowledge. This shows the flexibility of C-TeC in incorporating prior knowledge by narrowing the exploration space. Prior work does not offer this flexibility.

6.2 Exploration results (Q2)

In this experiment, we evaluate C-TeC in the `ant_large_maze`, `humanoid_u_maze`, and `arm_binpick_hard` environments. We run two variants of the experiment: (1) using the complete state vector as the future state, which is common in exploration tasks where the agent is encouraged to explore the entire state space; and (2) incorporating prior knowledge by narrowing the future state to specific components of the state vector. The latter allows us to assess whether C-TeC can flexibly explore subspaces of the state space, which is often useful in practice. In `ant_large_maze`, we define the future state as the future (x, y) position of the ant's torso. In `humanoid_u_maze`, we use the future (x, y, z) position of the humanoid's torso. Finally, in `arm_binpick_hard`, we define the future state as the future position of the cube.

As an evaluation metric, we count the number of unique discretized states covered by each agent. In `ant_large_maze`, we count the number of unique (x, y) positions in the maze visited by each agent. Similarly, in `humanoid_u_maze`, we count the number of visited (x, y, z) positions, and in `arm_binpick_hard`, we count the number of unique cube positions. We compare C-TeC to RND, ICM, APT, and a uniformly random policy. Figure 4 shows the learning curve when using the complete future state vector while Figure 5 shows the performance when we incorporate prior knowledge by restricting the future state to specific components of the state vector. . Each agent is run with 5 random seeds, and we plot the mean and standard deviation (Patterson et al., 2024).

Our agent outperforms the baselines in both variants of the experiment and learns interesting behaviors in the challenging `humanoid_u_maze` environment. Figure 6 shows screenshots of C-TeC behavior. More visuals are provided in Appendix G. This improvement can be the result of C-TeC's consistent reward properties. Methods like RND and ICM will eventually tend to zero reward as the state distribution is covered. A nice property of C-TeC is that it does not have zero reward in the limit.[1]

6.2.1 Learning complex behavior in Craftax-Classic

Can an RL policy learn complex behavior in Craftax-Classic without any task reward? To answer this question, we run C-TeC on Craftax-Classic (Matthews et al., 2024), a complex survival game where the agent's goal is to survive by crafting tools, maintaining food and shelter, and defeating enemies.

[1]Videos are in the project website: `https://temp-contrastive-explr.github.io/`

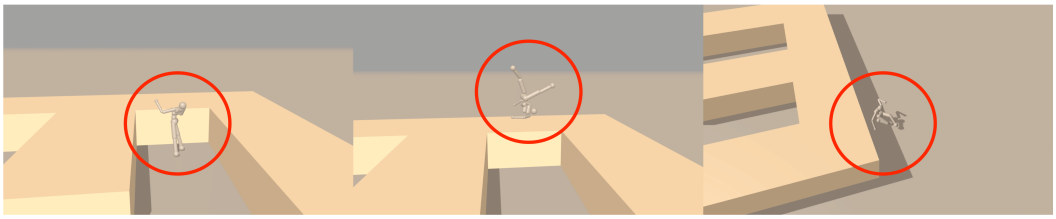

Figure 6: **C-TeC behavior in humanoid-u-maze**.C-TeC agent learns to escape the u-maze by jumping over the wall. None of the baseline intrinsic motivation methods discovered this kind of unexpected novel behavior.

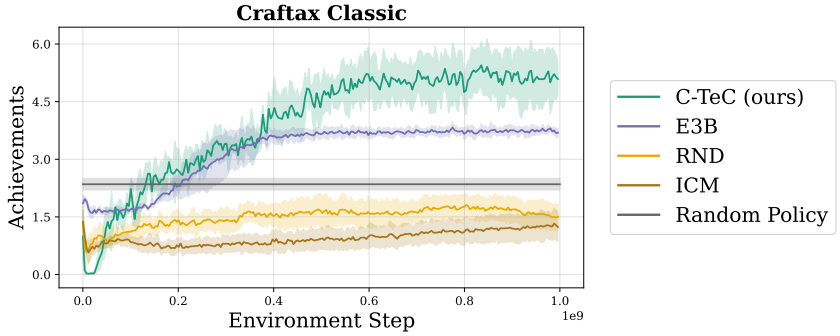

Figure 7: **Exploration in Craftax.** C-TeC outperforms the baselines in discovering more achievements in Craftax-Classic, E3B (Henaff et al., 2022) is the most competitive baseline.

In this experiment, we use the same PPO implementation as used in the baselines in the Craftax paper (Matthews et al., 2024), adding the C-TeC reward on top of it. We compare against RND, ICM, E3B, and a uniform random policy. We found that using PPO with memory (PPO-RNN) yields the best performance. The results are presented in Figure 7. The y-axis represents the sum of the achievements success rate, which measures how many capabilities and useful objects the agent has discovered. C-TeC outperforms the baselines and unlocks more achievements. Figure 18 visualizes some of the achievements of the C-TeC agent during an evaluation episode.

6.3 Sensitivity to future state sampling strategy (Q3)

In this experiment (see Figure 13), we investigate the sensitivity of C-TeC to the future state sampling strategy. Specifically, we consider two variants in addition to the geometric sampling. The first is uniformly sampling from the future. Unlike geometric sampling, uniform sampling does not prefer states that are sooner in the future over later ones. The second is geometric sampling with an increasing γ value. The intuition behind this strategy is that exploring nearby states is easier for the agent at the start of training, and as the agent becomes better at exploring them, it can progressively explore farther states in the future. We refer to this strategy as the γ-schedule, and we experiment with two different starting values of γ: one ranging from $\gamma = 0.9$ to $\gamma = 0.99$, and another from $\gamma = 0.1$ to $\gamma = 0.99$.

The results are shown in Figure 13. Regardless of the future state sampling strategy, the contrastive method explores better than the baselines in all three environments and appears robust.

7 Conclusion

This work has shown how to learn and leverage temporal contrastive representations for intrinsic exploration. With these representations, we construct a reward function that seeks out states with unexpected futures through the lens of representations. We find that C-TeC results in a significant performance gain over prior intrinsic objectives on state visitation metrics. These results hold over different RL algorithms and across environments. Future work includes further investigating the role of temporal representations for effective exploration.

Acknowledgements. We thank Marco Jiralerspong and Daniel Lawson for feedback on the draft of the paper. We thank Daniel Lawson, Roger Creus Castanyer, Siddarth Venkatraman, Raj Ghugare, Mahsa Bastankhah, and Grace Liu on discussions throughout the project. We thank Liv d'Aliberti for their plotting code and format for Figure 17. We thank the anonymous reviewers for helpful comments and feedback that improved the paper. We want to acknowledge funding support from Natural Sciences and Engineering Research Council of Canada, Samsung AI Lab, Google Research, Fonds de recherche du Québec, The Canadian Institute for Advanced Research (CIFAR), and IVADO. We acknowledge compute support from Digital Research Alliance of Canada, Mila IDT, and NVIDIA.

References

Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pp. 539–546. IEEE, 2005.

Empowerment: A universal agent-centric measure of control. In *2005 ieee congress on evolutionary computation*, volume 1, pp. 128–135. IEEE, 2005.

Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. *Advances in neural information processing systems*, 29, 2016.

Glen Berseth, Daniel Geng, Coline Manon Devin, Nicholas Rhinehart, Chelsea Finn, Dinesh Jayaraman, and Sergey Levine. {SM}irl: Surprise minimizing reinforcement learning in unstable environments. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=cPZOyoDloxl.

Martin Biehl, Christian Guckelsberger, Christoph Salge, C Smith, and Daniel Polani. Free energy, empowerment, and predictive information compared. Technical report, Technical report, University of Hertfordshire. URL: https://www. mis. mpg . . . , 2015.

Michał Bortkiewicz, Władysław Pałucki, Vivek Myers, Tadeusz Dziarmaga, Tomasz Arczewski, Łukasz Kuciński, and Benjamin Eysenbach. Accelerating goal-conditioned reinforcement learning algorithms and research. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=4gaySj8kvX.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation, 2018. URL https://arxiv.org/abs/1810.12894.

Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A. Efros. Large-scale study of curiosity-driven learning. In *International Conference on Learning Representations*, 2019a. URL https://openreview.net/forum?id=rJNwDjAqYX.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *International Conference on Learning Representations*, 2019b. URL https://openreview.net/forum?id=H1lJJnR5Ym.

Roger Creus Castanyer, Joshua Romoff, and Glen Berseth. Improving intrinsic exploration by creating stationary objectives. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=YbZxT0SON4.

Junya Chen, Zhe Gan, Xuan Li, Qing Guo, Liqun Chen, Shuyang Gao, Tagyoung Chung, Yi Xu, Belinda Zeng, Wenlian Lu, et al. Simpler, faster, stronger: Breaking the log-k curse on contrastive learners with flatnce. *CoRR*, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.

Thomas M. Cover and Joy A. Thomas. *Elements of information theory*. Wiley series in telecommunications. Wiley, New York, NY [u.a.], 1991. URL http://www.loc.gov/catdir/toc/onix06/90045484.html.

Ishan Dave, Rohit Gupta, Mamshad Nayeem Rizve, and Mubarak Shah. Tclr: Temporal contrastive learning for video representation. *Computer Vision and Image Understanding*, 219:103406, 2022.

Peter Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural computation*, 5(4):613–624, 1993.

Yilun Du, Chuang Gan, and Phillip Isola. Curious representation learning for embodied intelligence. ieee. In *CVF International Conference on Computer Vision*, pp. 10388–10397, 2021.

Yuqing Du, Stas Tiomkin, Emre Kiciman, Daniel Polani, Pieter Abbeel, and Anca Dragan. Ave: Assistance via empowerment. *Advances in Neural Information Processing Systems*, 33:4560–4571, 2020.

Benjamin Eysenbach, Ruslan Salakhutdinov, and Sergey Levine. C-learning: Learning to achieve goals via recursive classification. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=tc5qisoB-C.

Benjamin Eysenbach, Tianjun Zhang, Sergey Levine, and Ruslan Salakhutdinov. Contrastive learning as goal-conditioned reinforcement learning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=vGQiU5sqUe3.

Jesse Farebrother, Joshua Greaves, Rishabh Agarwal, Charline Le Lan, Ross Goroshin, Pablo Samuel Castro, and Marc G Bellemare. Proto-value networks: Scaling representation learning with auxiliary tasks. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=oGDKSt9JrZi.

Karl Friston. The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 11(2): 127–138, 2010.

Vincent Gardeux, Fabrice David, Adrian Shajkofci, Petra C. Schwalie, and Bart Deplancke. Asap: a web-based platform for the analysis and interactive visualization of single-cell rna-seq data. *Bioinformatics*, 33:3123 – 3125, 2016. URL https://api.semanticscholar.org/CorpusID:2237186.

Carles Gelada, Saurabh Kumar, Jacob Buckman, Ofir Nachum, and Marc G. Bellemare. DeepMDP: Learning continuous latent space models for representation learning. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2170–2179. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/gelada19a.html.

Lucas Gruaz, Alireza Modirshanechi, Sophia Becker, and Johanni Brea. Merits of curiosity: a simulation study. *PsyArXiv*, 2024.

Tuomas Haarnoja, Sehoon Ha, Aurick Zhou, Jie Tan, George Tucker, and Sergey Levine. Learning to walk via deep reinforcement learning. *arXiv preprint arXiv:1812.11103*, 2018a.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1861–1870. PMLR, 10–15 Jul 2018b. URL https://proceedings.mlr.press/v80/haarnoja18b.html.

Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018c.

Yusuke Hayashi and Koichi Takahashi. Universal ai maximizes variational empowerment. *arXiv preprint arXiv:2502.15820*, 2025.

Mikael Henaff, Roberta Raileanu, Minqi Jiang, and Tim Rocktäschel. Exploration via elliptical episodic bonuses. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=Xg-yZos9qJQ.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29, 2016.

Adriana Hugessen, Roger Creus Castanyer, Faisal Mohamed, and Glen Berseth. Surprise-adaptive intrinsic motivation for unsupervised reinforcement learning. *Reinforcement Learning Journal*, 2: 547–562, 2024.

Max Jaderberg, Volodymyr Mnih, Wojciech Marian Czarnecki, Tom Schaul, Joel Z Leibo, David Silver, and Koray Kavukcuoglu. Reinforcement learning with unsupervised auxiliary tasks. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=SJ6yPD5xg.

Tobias Jung, Daniel Polani, and Peter Stone. Empowerment for continuous agent—environment systems. *Adaptive Behavior*, 19(1):16–39, 2011.

Maximilian Karl, Philip Becker-Ehmck, Maximilian Soelch, Djalel Benbouzid, Patrick van der Smagt, and Justin Bayer. Unsupervised real-time control through variational empowerment. In *The International Symposium of Robotics Research*, pp. 158–173. Springer, 2019.

Alexander S Klyubin, Daniel Polani, and Chrystopher L Nehaniv. All else being equal be empowered. In *European Conference on Artificial Life*, pp. 744–753. Springer, 2005.

Michael Laskin, Aravind Srinivas, and Pieter Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. In *International conference on machine learning*, pp. 5639–5650. PMLR, 2020.

Michael Laskin, Denis Yarats, Hao Liu, Kimin Lee, Albert Zhan, Kevin Lu, Catherine Cang, Lerrel Pinto, and Pieter Abbeel. URLB: Unsupervised reinforcement learning benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=lwrPkQP_is.

Michael Laskin, Hao Liu, Xue Bin Peng, Denis Yarats, Aravind Rajeswaran, and Pieter Abbeel. Unsupervised reinforcement learning with contrastive intrinsic control, 2022. URL https://openreview.net/forum?id=9HBbWAsZxFt.

Lisa Lee, Benjamin Eysenbach, Emilio Parisotto, Eric P. Xing, Sergey Levine, and Ruslan Salakhutdinov. Efficient exploration via state marginal matching. *ArXiv*, abs/1906.05274, 2019. URL https://api.semanticscholar.org/CorpusID:186206676.

Andrew Levy, Alessandro Allievi, and George Konidaris. Latent-predictive empowerment: Measuring empowerment without a simulator. *arXiv preprint arXiv:2410.11155*, 2024.

Grace Liu, Michael Tang, and Benjamin Eysenbach. A single goal is all you need: Skills and exploration emerge from contrastive RL without rewards, demonstrations, or subgoals. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=xCkgX4Xfu0.

Hao Liu and Pieter Abbeel. Behavior from the void: Unsupervised active pre-training. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=fIn4wLS2XzU.

Zhuang Ma and Michael Collins. Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 3698–3707, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. DOI: 10.18653/v1/D18-1405. URL https://aclanthology.org/D18-1405/.

Marlos C Machado, Marc G Bellemare, and Michael Bowling. Count-based exploration with the successor representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 5125–5133, 2020.

Jarryd Martin, Suraj Narayanan Sasikumar, Tom Everitt, and Marcus Hutter. Count-based exploration in feature space for reinforcement learning. *arXiv preprint arXiv:1706.08090*, 2017.

Michael Matthews, Michael Beukman, Benjamin Ellis, Mikayel Samvelyan, Matthew Thomas Jackson, Samuel Coward, and Jakob Nicolaus Foerster. Craftax: A lightning-fast benchmark for open-ended reinforcement learning. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=hg4wXlrQCV.

Bogdan Mazoure, Benjamin Eysenbach, Ofir Nachum, and Jonathan Tompson. Contrastive value learning: Implicit models for simple offline rl. In *Conference on Robot Learning*, pp. 1257–1267. PMLR, 2023.

Shakir Mohamed and Danilo Jimenez Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. *Advances in neural information processing systems*, 28, 2015.

Vivek Myers, Evan Ellis, Sergey Levine, Benjamin Eysenbach, and Anca Dragan. Learning to assist humans without inferring rewards. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=WCnJmb7cv1.

Alexander Nikulin, Vladislav Kurenkov, Ilya Zisman, Artem Agarkov, Viacheslav Sinii, and Sergey Kolesnikov. Xland-minigrid: Scalable meta-reinforcement learning environments in jax. *Advances in Neural Information Processing Systems*, 37:43809–43835, 2024.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Georg Ostrovski, Marc G Bellemare, Aäron Oord, and Rémi Munos. Count-based exploration with neural density models. In *International conference on machine learning*, pp. 2721–2730. PMLR, 2017.

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.

Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. Self-supervised exploration via disagreement, 2019. URL https://arxiv.org/abs/1906.04161.

Andrew Patterson, Samuel Neumann, Martha White, and Adam White. Empirical design in reinforcement learning. *Journal of Machine Learning Research*, 25(318):1–63, 2024. URL http://jmlr.org/papers/v25/23-0183.html.

Silviu Pitis, Harris Chan, Stephen Zhao, Bradly Stadie, and Jimmy Ba. Maximum entropy gain exploration for long horizon multi-goal reinforcement learning, 2020. URL https://arxiv.org/abs/2007.02832.

Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6964–6974, 2021.

Nicholas Rhinehart, Jenny Wang, Glen Berseth, John D Co-Reyes, Danijar Hafner, Chelsea Finn, and Sergey Levine. Information is power: Intrinsic control via information capture. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=MO76tBOz9RL.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Max Schwarzer, Ankesh Anand, Rishab Goel, R Devon Hjelm, Aaron Courville, and Philip Bachman. Data-efficient reinforcement learning with self-predictive representations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=uCQfPZwRaUu.

Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, Sergey Levine, and Google Brain. Time-contrastive networks: Self-supervised learning from video. In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 1134–1141. IEEE, 2018.

Bradly C Stadie, Sergey Levine, and Pieter Abbeel. Incentivizing exploration in reinforcement learning with deep predictive models. *arXiv preprint arXiv:1507.00814*, 2015.

Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. # exploration: A study of count-based exploration for deep reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

Sebastian B Thrun. *Efficient exploration in reinforcement learning*. Carnegie Mellon University, 1992.

Ahmed Touati and Yann Ollivier. Learning one representation to optimize all rewards. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=2a96Bf7Qdrg.

Ahmed Touati, Jérémy Rapin, and Yann Ollivier. Does zero-shot reinforcement learning exist? In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=MYEap_OcQI.

Zhi-Xiong Xu, Xi-Liang Chen, Lei Cao, and Chen-Xi Li. A study of count-based exploration and bonus for reinforcement learning. In *2017 IEEE 2nd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, pp. 425–429. IEEE, 2017.

Rushuai Yang, Chenjia Bai, Hongyi Guo, Siyuan Li, Bin Zhao, Zhen Wang, Peng Liu, and Xuelong Li. Behavior contrastive learning for unsupervised skill discovery. In *International conference on machine learning*, pp. 39183–39204. PMLR, 2023.

Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Reinforcement learning with prototypical representations. In *International Conference on Machine Learning*, pp. 11920–11931. PMLR, 2021.

Amy Zhang, Rowan Thomas McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=-2FCwDKRREu.

Ruihan Zhao, Kevin Lu, Pieter Abbeel, and Stas Tiomkin. Efficient empowerment estimation for unsupervised stabilization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=u2YNJPcQlwq.

Chongyi Zheng, Jens Tuyls, Joanne Peng, and Benjamin Eysenbach. Can a MISL fly? analysis and ingredients for mutual information skill learning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=xoIeVdFO7U.

Supplementary Materials

The following content was not necessarily subject to peer review.

Broader Impact Statement

This work proposes an exploration method for deep RL agents that facilitates finding better solutions across a broad range of sequential decision-making problems. Depending on the intended task and the reward function, the resulting policy may lead to either positive or negative consequences.

A Training Details and Ablations

We summarize the hyperparameters and model architectures for all experiments. In Appendix A.1, we provide the training details for the locomotion and manipulation experiments. In Appendix A.2, we provide the details of the Craftax experiments. In Appendix A.3, we provide the details of all the environments. In Appendix A.5, we include the codebase.

Finally, in Appendix C, we include the ablation experiments.

A.1 Robotics Environments

In the robotics environments, we used SAC as the RL algorithm. Table 1 shows the hyperparameters that are shared across all methods.Table 2 and Table 3 show the algorithm-specific hyperparameters for C-TeC and the baselines, respectively.

Table 1: Hyperparameters for all methods in robotics environments

Hyperparameter	Value
num_timesteps	500,000,000
max_replay_size	10,000
min_replay_size	1,000
episode_length	1,000
discounting	0.99
num_envs	1024 (256 for humanoid_u_maze)
batch_size	1024 (256 for humanoid_u_maze)
multiplier_num_sgd_steps	1
action_repeat	1
unroll_length	62
policy_lr	3e-4
critic_lr	3e-4
hidden layers (for both actor and critic)	[256,256]

Table 2: Hyperparameters for C-TeC in robotics environments

Hyperparameter	Value
contrastive_lr	3e-4
contrastive_loss_function	InfoNCE
similarity_function	L1
logsumexp_penalty	0.1
hidden layers (for both encoders)	[1024,1024]
representation dimension	64

Table 3: Hyperparameters for baselines in robotics environments

Hyperparameter	Value
rnd encoder lr	3e-4
rnd embedding dim	512
rnd encoder hidden layers	[256, 256]
icm encoder lr (forward and inverse models)	3e-4
icm embeddings_dim	512
icm encoders hidden layers	[1024, 1024]
icm weight on forward loss	0.2
apt contrastive lr	3e-4
apt similarity function	L1
apt contrastive hidden layers	[1024, 1024]
apt representation dimension	64
Augmentation type	$\mathcal{N}(0, 0.5)$

A.2 Craftax

In Craftax, we used PPO as the RL algorithm[2]. Table 4 shows the hyperparameters shared across all methods. Table 5 and Table 6 show the algorithm-specific hyperparameters for C-TeC and the baselines, respectively.

Table 4: Hyperparameters for all methods in robotics environments

Hyperparameter	Value
num_timesteps	1,000,000,000
num_steps	64
learning_rate	2e-4
anneal_learning_rate	True
update_epochs	4
discounting	0.99
gae_lambda	0.8
clip_epsilon	0.2
ent_coef	0.01
max_grad_norm	1.0
activation	tanh
action_repeat	1
RNN_layers (GRU)	[512 (embedding dim),512 (hidden dim)]
hidden layers (both actor and value)	[512, 512]

[2] https://github.com/MichaelTMatthews/Craftax_Baselines

Table 5: Hyperparameters for C-TeC in Craftax

Hyperparameter	Value
contrastive_lr	3e-4
contrastive_loss_function	InfoNCE
similarity_function	L2
logsumexp_penalty	0.0
hidden layers (for both encoders)	[1024,1024,1024]
representation dimension	64

Table 6: Hyperparameters for baselines in Craftax

Hyperparameter	Value
rnd encoder lr	3e-4
rnd embedding dim	512
rnd encoder hidden layers	[256, 256]
icm encoder lr (forward and inverse models)	3e-4
icm embeddings_dim	512
icm encoders hidden layers	[256, 256]
icm weight on forward loss	1.0
e3b (icm) lambda	0.1

A.3 Environment Details

- **Ant-hardest-maze** The observation space of this environment has 29 dimensions, consisting of joint angles, angular velocities, and the x,y position of the ant's torso. The action space is 7-dimensional, representing the torque applied to each joint.
- **Humanoid-u-maze** The observation space of this environment has 268 dimensions, consisting of joint angles, angular velocities, and the x,y position of the humanoid's torso. The action space is 17-dimensional, representing the torque applied to each joint.
- **Arm-binpick-hard** The observation space of this environment has 18 dimensions, consisting of joint angles, angular velocities, the cube position, and the end-effector position and offset. The action space is 5-dimensional, representing the displacement of the end-effector.
- **Craftax-Classic** The observation space is a one-hot encoding of size 1345, capturing player information (inventory, health, hunger, attributes, etc.) as well as the types of blocks and creatures within the player's visual field. The action space is discrete and consists of 17 actions.

A.4 Details on C-TeC reward

One important detail is that the policy's objective is slightly different from the (negative) representation objective (Equation (13)) because it omits the log-sum-exp term. This can be seen by rewriting the reward function as follows:

$$r_{\text{intr}}(s,a) = \mathbb{E}_{p_{\mathcal{T}}(s_f|s,a)}[\underbrace{\|\phi(s,a) - \psi(s_f)\| + \log \sum_{s'_f} e^{-\|\phi(s,a)-\psi(s'_f)\|}}_{\text{(neg) contrastive loss}} - \log \sum_{s'_f} e^{-\|\phi(s,a)-\psi(s'_f)\|}]. \tag{8}$$

To further gain intuition for what this is doing, we note that (in practice) the $\phi(s,a)$ representations are quite similar to the $\psi(s)$ representation evaluated at the same state. Thus, we can approximate

this second term as

$$\log \sum_{s'_f} e^{-\|\psi(s)-\psi(s'_f)\|} \approx \log \hat{p}(s), \tag{9}$$

which we identify as a kernel density estimate of the marginal likelihood of state s under the replay buffer distribution $p_{\mathcal{T}}(s)$. This observation helps explain why including the log-sum-exp term in the reward would degrade performance – it effectively corresponds to *minimizing* state entropy, which can often hinder exploration, especially in environments without much noise (Zheng et al., 2025). One additional consideration here is that, because the likelihood is measured using learned representations, it is sensitive to the policy's understanding of the environment. While ordinarily maximizing state entropy can lead to degenerate solutions (like the noisy TV), our approach mitigates this problem because the contrastive representations will only learn features that are predictive of future states (hence, they would ignore a noisy TV).

A.4.1 Variance reduction in the reward estimate

We can decrease the variance in our estimate of the expectation in Equation (5) by looking at all future states $s_f = s_{t+1}, s_{t+2}, \cdots$ and weighting each summand by γ^i:

$$\begin{aligned} r_t &= \mathbb{E}_{p(s_f|s_t,a_t)}[r_{\text{int}}(s_t, a_t, s_f)] && (10)\\ &= \mathbb{E}_{p(s_f|s_t,a_t)}[||\phi(s,a) - \psi(s_{t'})||_2] && (11)\\ &\approx \frac{1-\gamma^{H-t}}{1-\gamma} \sum_{t'=t}^{H} \gamma^{t'-t} ||\phi(s,a) - \psi(s_{t'})||_2 && (12) \end{aligned}$$

The (unbiased) approximation comes because we only look at future states that occur in one trajectory, and other trajectories might visit different future states. The ugly fraction is the normalizing constant for a truncated geometric series. In the last line, note that the summation $\sum_{t'=t}^{H} \gamma^{t'-t}\psi(s_{t'})$ can be quickly computed for every r_t by starting at $T = H$ and decrementing t, updating $\psi_{\text{sum}} = \psi(s_t) + \gamma\psi_{\text{sum}}$. This is the same trick that's usually used for computing the empirical future returns in REINFORCE, and decreases compute from $\mathcal{O}(H^2)$ to $\mathcal{O}(H)$. We use this estimator in Craftax-Classic but we found that omitting the normalization term results in much better performance.

A.5 Codebase

Our codebase for the robotics experiments and Craftax is provided below:

- **Robotics Environments** https://github.com/FaisalAhmed0/c-tec
- **Craftax** https://github.com/FaisalAhmed0/c-tec/tree/craftax

B Compute Resources

In all experiments, we use 2 CPUs, a single GPU, and 8 GB of RAM. The specific GPU type varies depending on the job scheduling system, but most experiments run on NVIDIA RTX 8000 or V100 GPUs. Training in the robotics environments takes approximately 24 hours on average, while Craftax experiments require around 30 hours.

C Ablation Study

To understand the contribution of each component to the overall performance of C-TeC, we conduct an ablation study on several key elements of the algorithm, illustraed in the following section.

C.1 Representation Normalization

Is it important to normalize the contrastive representations when computing the intrinsic reward? To answer this question, we compare the exploration performance of C-TeC across all environments, keeping all hyperparameters fixed except for the normalization of the representations.

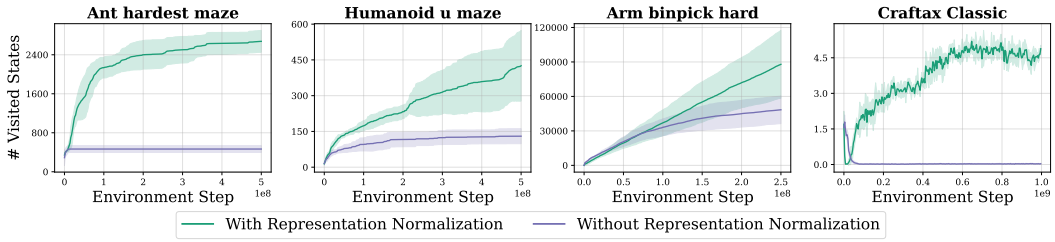

Figure 8: **Normalizing the contrastive representations.** Normalizing the representations is crucial for effective exploration—using unnormalized representations significantly degrades exploration performance.

C.2 Contrastive Losses

We compare the performance of C-TeC using different contrastive loss functions. Specifically, we evaluate InfoNCE, symmetric InfoNCE, NCE (Hjelm et al.), FlatNCE (Chen et al., 2021), and a Monte-Carlo version of the forward-backward (FB) (Touati & Ollivier, 2021) loss, as defined in [Equation (13)–Equation (17)]. Figure 9 presents the results. Overall, NCE leads to poorer exploration, particularly in Craftax. InfoNCE and symmetric InfoNCE exhibit similar performance across all environments. In general, the method is reasonably robust to the choice of contrastive loss.

$$\mathcal{L}_{\text{InfoNCE}}(\theta) = -\sum_{i=1}^{K} \log \left(\frac{e^{C_\theta((s_i,a_i),s_f^{(i)})}}{\sum_{j=1}^{K} e^{C_\theta((s_i,a_i),s_f^{(j)})}} \right) \tag{13}$$

$$\mathcal{L}_{\text{symmetric\_InfoNCE}}(\theta) = -\left[\sum_{i=1}^{K} \log \left(\frac{e^{C_\theta((s_i,a_i),s_f^{(i)})}}{\sum_{j=1}^{K} e^{C_\theta((s_i,a_i),s_f^{(j)})}} \right) + \log \left(\frac{e^{C_\theta((s_i,a_i),s_f^{(i)})}}{\sum_{j=1}^{K} e^{C_\theta((s_j,a_j),s_f^{(i)})}} \right) \right] \tag{14}$$

$$\mathcal{L}_{\text{Binary(NCE)}}(\theta) = -\left[\sum_{i=1}^{K} \log \left(\sigma \left(C_\theta((s_i,a_i),s_f^{(i)}) \right) \right) - \sum_{j=2}^{K} \log \left(1 - \sigma \left(C_\theta((s_i,a_i),s_f^{(j)}) \right) \right) \right] \tag{15}$$

$$\mathcal{L}_{\text{FlatNCE}}(\theta) = -\sum_{i=1}^{K} \log \left(\frac{\sum_{j=1}^{K} e^{C_\theta(s_i,a_i,s_f^{(j)}) - C_\theta(s_i,a_i,s_f^{(i)})}}{\text{detach}\left[\sum_{j=1}^{K} e^{C_\theta(s_i,a_i,s_f^{(j)}) - C_\theta(s_i,a_i,s_f^{(i)})} \right]} \right) \tag{16}$$

$$\mathcal{L}_{\text{FB}}(\theta) = -\sum_{i=1}^{K} \left(e^{C_\theta(s_i,a_i,s_f^{(i)})} \right) + \frac{1}{2(K-1)} \sum_{i=1}^{K} \sum_{\substack{j=1 \\ j \neq i}}^{K} \left(e^{C_\theta(s_i,a_i,s_f^{(j)})} \right)^2 \tag{17}$$

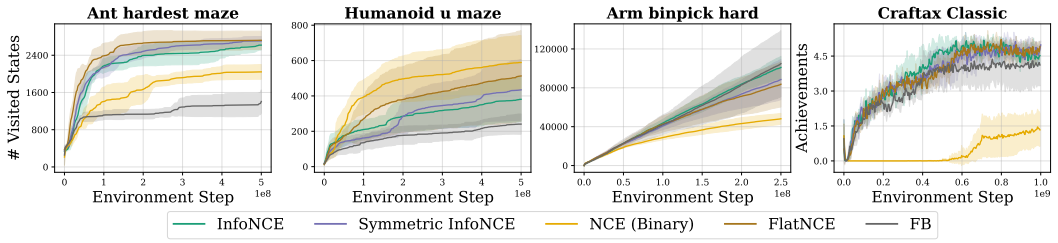

Figure 9: **Comparison of Different Contrastive Losses.** Overall, C-TeC is robust to the choice of contrastive loss. A notable exception is the Binary NCE loss in Craftax, where it performs relatively poorly.

C.3 Contrastive Critic Functions

We compare four critic similarity functions shown below:

$$C_\theta((s_t, a_t), s_f)_{L1} = -||\phi_\theta(s_t, a_t) - \psi_\theta(s_f)||_1. \tag{18}$$

$$C_\theta((s_t, a_t), s_f)_{L2} = -||\phi_\theta(s_t, a_t) - \psi_\theta(s_f)||_2 \tag{19}$$

$$C_\theta((s_t, a_t), s_f)_{L2-w/o-sqrt} = -||\phi_\theta(s_t, a_t) - \psi_\theta(s_f)||_2^2 \tag{20}$$

$$C_\theta((s_t, a_t), s_f)_{dot} = -\phi_\theta(s_t, a_t)^\top \psi_\theta(s_f) \tag{21}$$

Figure 14 shows the results. In general, using the L_1 distance yields the best performance across the robotic environments, while L_2 performs better in Craftax. This highlights the importance of this design choice and suggests that some tuning may be required to select the most effective critic function.

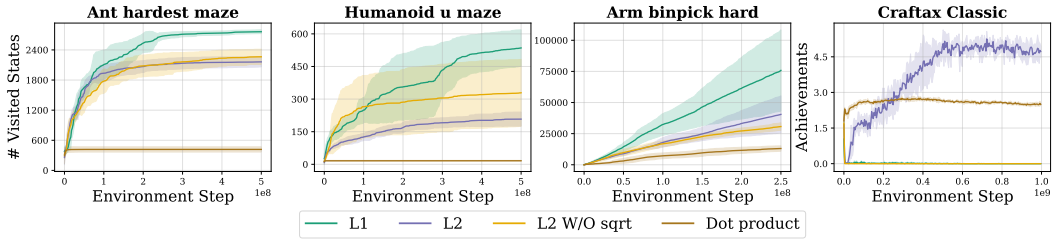

Figure 10: **Comparison of Critic function.** Overall,the L_1 distance yields the best performance across the robotic environments, while L_2 performs better in Craftax.

C.4 Contrastive Critic Architecture

In this ablation we compare two architectures of the contrastive critic, the separable architecture $(\phi_\theta(s_t, a_t), \psi_\theta(s_f))$, which is the one we use in all of our experiment, and the monolithic critic f_θ i.e., a single model that takes in triplet $f(s, a, s_f)$.

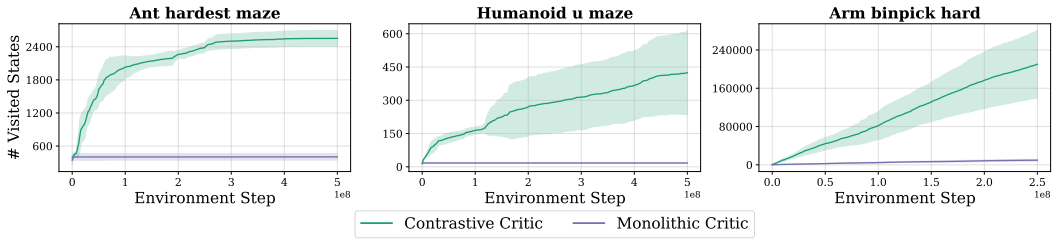

Figure 11: **Critic parameterization** Using a monolithic critic results in poor exploration performance, while using the separable architecture results in much better exploration. This shows the importance of the critic parameterization as distance function between two representations.

Importantly, these experiments show that the factorized representation parameterization is a necessary (relative to the monolithic critic) condition for effective exploration. We discuss the possible failure mode of using the monolithic critic in Section 5.2. These experiments do not demonstrate sufficiency, and we claim that the information-theoretic interpretation for a critic that fully captures the point-wise MI is still useful for analysis.

C.5 Forward vs. reverse KL

As mentioned in Section 5.1, we hypothesized that the reverse KL C-TeC reward is important for exploration. As it encourages mode-seeking behavior (prioritizing unfamiliar states), to test this hypothesis we run C-TeC with the negative-forward KL reward (**??**), the results shown in Appendix C.5 indicates that using the reverse is necessary for exploration.

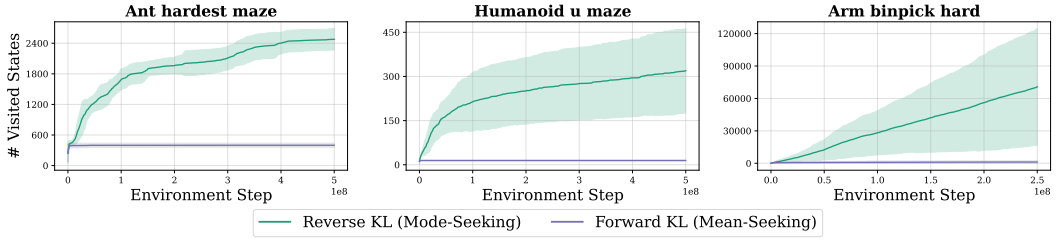

Figure 12: **Forward vs Reverse KL** C-TeC with the reverse KL reward promotes mode-seeking behavior which encourages the agent to prioritize visiting unfamiliar states resulting in much better exploration.

C.6 Future state sampling strategy

While the results are robust across sampling strategy, the best-performing strategy is environment-specific. For example, in `ant_hardest_maze`, sampling according to the γ-schedule performs best, while in `arm_binpick_hard`, geometric sampling tends to perform slightly better.

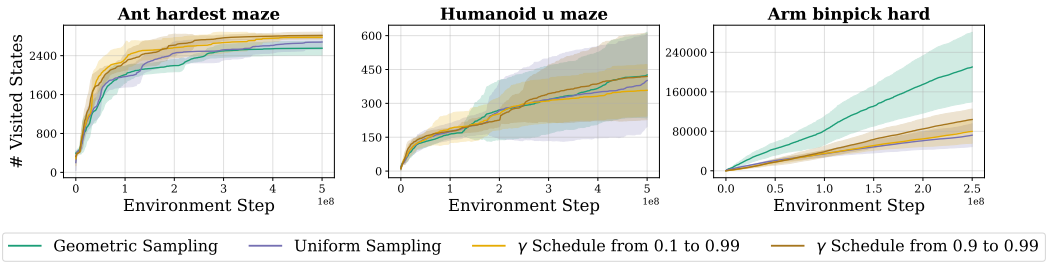

Figure 13: **Sensitivity to future state sampling strategy.** We compare variants of C-TeC with different future state sampling strategy, the method is robust to the choice of the sampling strategy and all the variants outperform the baselines.

D Exploration in Noisy TV setting

We investigate C-TeC performance in the presence of a noisy TV state, we run this experiment on a modified grid environment from xland-minigrid (Nikulin et al., 2024) of size 256×256 Figure 15 with a noisy TV region. We did not observe any evidences of worse exploration performance namely the agent has covered all the states in the grid world, Appendix D shows the state coverage of C-TeC compared to the maximum coverage.

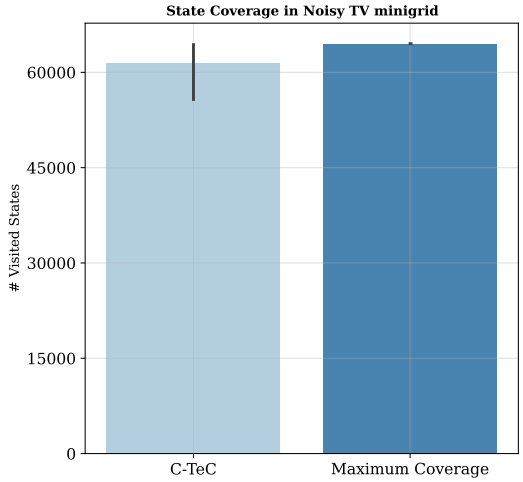

Figure 14: **C-TeC Coverage in noisy TV setting** C-TeC can effectively explore in the presence of noisy states

E Comparison with Previous Methods

At a high level, C-TeC is related to other intrinsic exploration objectives that reward uncertainty. Objectives such as RND (Burda et al., 2019b) and Disagreement (Pathak et al., 2019) explore unfamiliar states, presumably leading to these states becoming more familiar in future rounds. A related method, CURL (Du et al., 2021), also relies on using a negative contrastive similarity score for exploration like C-TeC. CURL prioritizes exploration over states with high error/low similarity scores with augmented states; however, the contrastive features learned in CURL are not temporal and can be concretely related to prediction error.

The key difference between prior methods and C-TeC lies in the usage of temporal contrastive features. Our method drives the agent to explore areas where *future* outcomes have been seen but appear

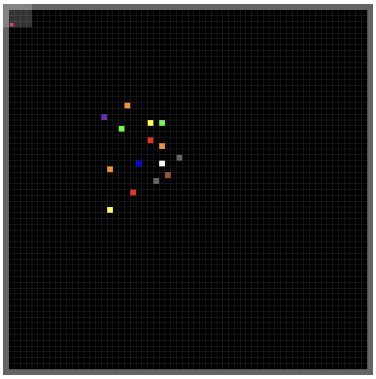

Figure 15: Xland-Minigrid (Nikulin et al., 2024) with noisy TV states indicated by the random colors.

improbable. Taken together, our analysis and results show that temporal contrastive representations are simple yet powerful frameworks for intrinsic motivation.

F Intrinsic Reward Interpretation

On information-theoretic interpretation of reward. The intrinsic reward with representations rewards (s, a) pairs that result in the largest additional number of bits needed to encode the representation induced $p_{\phi,\psi}(s_f \mid s_t, a_t)$ with a code optimized for the marginal $p_{\phi,\psi}(s_f)$. In other words, it prioritizes exploration in areas where the representation encoding schemes is highly inefficient.

On information-theoretic interpretation of objective assuming perfect estimation of point-wise MI. We assume that representations perfectly capture point-wise MI. Taking an additional expectation of the roll-out state-occupancy reveals that the PPO/SAC objective is a minimization of MI

$$J^{\pi} = \mathbb{E}_{p_\pi(s,a), p_\mathcal{T}(s_f|s,a)}[r_{\text{intr}}(s_t, a_t)] \approx -I[S_f; S_\pi, A_\pi] \tag{22}$$

where p_π is the policy induced discounted state-occupancy measure (see Eq. 1).

On C-TeC as a Two-Player Game In addition to quantifying temporal similarity, the converged InfoNCE loss $\mathcal{L}^*_{\text{CRL}}$ provides a lower bound on the mutual information (MI) (Oord et al., 2018; Eysenbach et al., 2021):

$$I(S_f; S_t, A_t) \geq \log K - \mathcal{L}^*_{\text{CRL}}(\mathcal{B}; \theta).$$

Contrastive learning finds representations that maximize a lower bound on the MI between *current* states and actions and *future* state distributions.

Thus, we can view C-TeC as a two-player game over an ever-expanding buffer. Namely, the CL step learns to minimize $\mathcal{L}_{\text{CRL}}$. Meanwhile, the policy objective learns to approximately maximize $\mathcal{L}_{\text{CRL}}$ when state-action pairs are strictly drawn from the roll-out policy (as opposed to the entire buffer), and the conditional and marginal future-state distributions are still defined over the buffer.

F.1 No (Achievable) Trivial Fixed Points

Does C-TeC have stable fixed points? Without additional simplifications, this problem is intractable. Notably, standard analysis would fail to prove convergence due to the non-convexity/concavity of the objectives. While the zero-gradient condition for the InfoNCE objective is clear, the zero-gradient condition for the objective is not obvious due to the complex relationship between π and the state occupancy measure.

A more aggressive simplification that can simplify analysis of the global optimum is to (1) assume that the policy optimization is done directly over S_π and A_π and (2) assume the representations perfectly capture the point-wise MI. Furthermore, we assume that future states are exclusively sampled from the one-step transition dynamics and are deterministic.

In practice, these assumptions are very unrealistic; however, such simplifications have been used in prior work on unsupervised RL to give a conceptual picture of exploration methods (Pitis et al., 2020). Throughout, we assume fully expressive representations that capture the point-wise MI – thus, we are strictly analyzing fixed points and fixed point-stability/achievability without taking into account representations.

Though the following analysis assumes a discrete setting (summations vs. integrals, Kronecker Deltas vs. Dirac Deltas), we do not directly invoke the assumption of discreteness. The conclusions should continue to hold in the continuous case assuming all relevant probability distributions are bounded and smooth.

With these simplifications, the InfoNCE objective reduces to:

$$\max_{\phi,\psi} \; [\log K - \mathcal{L}_{\phi,\psi}(Z_\mathcal{T}, F_\mathcal{T})] \underset{K\to\infty, \text{infinitely expressive reps}}{\longrightarrow} I\big(S_\mathcal{T}, A_\mathcal{T};\, S_f\big).$$

Because the "policy" optimization is fixed in $p(s_f \mid s, a)$, the MI $I(S_f; S_\pi, A_\pi)$ (see Eq. 22) is concave in $p_\pi(s, a)$ and $p_\pi(s_f)$ (Cover & Thomas, 1991). Our objective has now reduced to a constrained optimization problem with conditions $\sum_{s,a} p_\pi(s, a) = 1$ and $p_\pi(s, a) \geq 0$ for all $(s, a) \in \mathcal{S} \times \mathcal{A}$.

Consider the fixed point conditions given by the Lagrangian that is Lipschitz-continuous over the probability simplex $\Delta_\mathcal{S}$. Let λ and $\mu(s, a)$ denote the Lagrange multipliers for the normalization and non-negativity conditions respectively. Then, the full Lagrangian $\mathcal{L}_{\text{Lagrangian}}$ is as follows:

$$\mathcal{L}_{\text{Lagrangian}}(p_\pi, \lambda, \mu) = I(S_\pi, A_\pi; S_f) + \lambda\Big(\sum_{s,a} p_\pi(s, a) - 1\Big) - \sum_{s,a} \mu(s, a) p_\pi(s, a).$$

Note that by complementary slackness, we have $\mu(s, a)p(s, a) = 0$. Taking the functional derivative of $\mathcal{L}_{\text{Lagrangian}}$ with respect to distribution $p(s, a)$ yields the KL-divergence:

$$\frac{\delta \mathcal{L}_{\text{Lagrangian}}}{\delta p_\pi}[s, a] = D_{KL}[p_\mathcal{T}(s_f \mid s, a) || p_\mathcal{T}(s_f)] - 1 + \lambda - \mu(s, a).$$

By the complementary slackness, the distribution $p_\pi(s, a)$ is a fixed point if the KL-divergence $D_{KL}[p_\mathcal{T}(s_f \mid s, a) || p_\mathcal{T}(s_f)]$ is *constant* for any (s, a) where $p_\pi(s, a)$ has support. Any deviation would lead to a non-zero gradient at the point (s, a). In other words, all conditional trajectory future state distributions look equally "far" from the marginal.

Stationarity over *iterations* of C-TeC requires an additional condition: that the D_{KL} remains constant over all (s, a) *after* updating buffer $p_\mathcal{T}(s_f)$ with states encountered during the roll-out. A model of this is reweighing the marginal with the rollout probability distribution $p_\pi(s_f) = \sum_{s,a} p_\pi(s_f \mid s, a) p_\pi(s, a)$:

$$D_{KL}[p_\mathcal{T}(s_f \mid s, a) || p'(s_f)] = D_{KL}[p_\mathcal{T}(s_f \mid s, a) || (1 - \alpha) \cdot p_\mathcal{T}(s_f) + \alpha \cdot p_\pi(s_f)] \tag{23}$$

where $0 < \alpha < 1$. Again, we assume deterministic dynamics for simplicity and that s_f is always the next state (i.e. small discount factor like the Craftax setting) so the conditional distribution does not change. We have no easy way of determining the change in $p_\mathcal{T}(s_f \mid s, a)$ after roll-out.

We drop subscripts on transitions and simplify:

$$\begin{aligned}
LHS &= \sum_{s_f} p(s_f \mid s,a) \log \frac{p(s_f \mid s,a)}{p_{\mathcal{T}}(s_f)} - \sum_{s_f} p(s_f \mid s,a) \log \frac{p_{\mathcal{T}}(s_f)}{(1-\alpha)\cdot p_{\mathcal{T}}(s_f) + \alpha \cdot p_\pi(s_f)} \\
&= D_{KL}[p_{\mathcal{T}}(s_f \mid s,a) || p_{\mathcal{T}}(s_f)] - \mathbb{E}_{p(s_f|s,a)}\Big[\log \frac{p_{\mathcal{T}}(s_f)}{(1-\alpha)\cdot p_{\mathcal{T}}(s_f) + \alpha \cdot p_\pi(s_f)}\Big] \\
&= C_{\text{old}} - \mathbb{E}_{p(s_f|s,a)}\Big[\log \frac{p_{\mathcal{T}}(s_f)}{(1-\alpha)\cdot p_{\mathcal{T}}(s_f) + \alpha \cdot p_\pi(s_f)}\Big]
\end{aligned}$$

where C_{old} is the old constant D_{KL} across (s,a). Thus, for the LHS to also be constant across (s,a), the difference must also be constant. We assume that transitions are nontrivial (as in, $p(s_f \mid s,a) \neq p(s_f)$). This implies that the updated D_{KL} remains constant iff

$$\begin{aligned}
(1-\alpha)\cdot p_{\mathcal{T}}(s_f) + \alpha \cdot p_\pi(s_f) &= p_{\mathcal{T}}(s_f) \\
\Rightarrow p_{\mathcal{T}}(s_f) &= p_\pi(s_f).
\end{aligned}$$

Under the assumptions of one-step, deterministic transitions and the α-reweighing of the buffer distribution, the distribution $p_\pi(s,a)$ remains a fixed point iff the roll-out future distribution and buffer future distribution are identical.

What is the stability of these fixed points? We can do linear fixed-point stability analysis by calculating the Jacobian of the update, where prime (') denotes the next-step $\delta p_\pi(s_f)$. The update of $\delta p_\pi(s_f)$ is as follows:

$$\begin{aligned}
\delta p'_\pi(s_f) &= \delta p_\pi(s_f) - \eta p(s_f \mid s,a)\Big[\Big(\nabla^2_{p_\pi(s,a)} I(S_\pi, A_\pi; S_f)\Big)\, \delta p_\pi\Big](s,a) && (24) \\
&= \delta p_\pi(s_f) - \eta\Big[\Big(\nabla^2_{p_\pi(s_f)} I(S_\pi, A_\pi; S_f)\Big)\, \delta p_\pi\Big](s_f) && \text{(change of vars.)} \\
&= \Big(I - \eta \nabla^2_{p_\pi(s_f)} I(S_\pi, A_\pi; S_f)\Big)\delta p_\pi(s_f), && (25)
\end{aligned}$$

We can similarly calculate the update for $\delta p_{\mathcal{T}}(s_f)$:

$$\begin{aligned}
\delta p'_{\mathcal{T}}(s_f) = \alpha\Big(I - \eta \nabla^2_{p_\pi(s_f)} I(S_\pi, A_\pi; S_f)\Big)\, \delta p_\pi(s_f) && \text{(weight new traj.)} \\
+ (1-\alpha)\delta p_{\mathcal{T}}(s_f). && \text{(down-weight old traj.)}
\end{aligned}$$

Thus, the equation relating $(\delta p_\pi(s_f), \delta p_{\mathcal{T}}(s_f))$ and $(\delta p'_\pi(s_f), \delta p'_{\mathcal{T}}(s_f))$ is

$$\begin{pmatrix} \delta p'_\pi(s_f) \\ \delta p'_{\mathcal{T}}(s_f) \end{pmatrix} = \underbrace{\begin{pmatrix} I - \eta H & 0 \\ \alpha\,(I - \eta H) & (1-\alpha)\, I \end{pmatrix}}_{J} \begin{pmatrix} \delta p_\pi(s_f) \\ \delta p_{\mathcal{T}}(s_f) \end{pmatrix}.$$

to first order in iteration time τ, where H is the Hessian of the MI with respect to $p_\pi(s_f)$. Because the MI is concave in $p_\pi(s_f)$, the Hessian H is negative semi-definite; note that if H has any negative eigenvalues at the fixed point, the Jacobian would have at least one eigenvalue > 1. Thus, the non-vertex fixed points in the product of two probability simplices $\Delta_{\mathcal{S}} \times \Delta_{\mathcal{S}}$ (where $p_{\mathcal{T}} = p_\pi(s_f)$) are either unstable, where at least one direction corresponds to an eigenvalue > 1 in the Jacobian, or semi-stable fixed points, where the MI is locally flat at the fixed point. Finally, fixed points at the vertices of the probability simplex (Delta functions) are uninteresting and are not observed in practice.

For an arbitrary MDP, we note that semi-stable fixed points are generally hard to achieve: a nontrivial, non-constant transition function, random roll-outs at initialization, the mixture of policies in the buffer, and newly encountered states prevent such semi-stable states from being easily accessible. Particularly, the random roll-outs help prevent no-op from being a trivial fixed point.

This analysis shows that there are no easily-obtainable, stable fixed points for standard MDPs even under aggressive simplifications, implying constantly evolving probability distributions. Future work remains to investigate the existence of dynamical steady-states and whether the reached probability distributions cover a large region of the probability simplex.

G Emergent Exploration Behavior

Figure 17 shows some of the learned behaviors of C-TeC in the `humanoid-u-maze`, where the agent learns to jump over the wall to escape the maze.

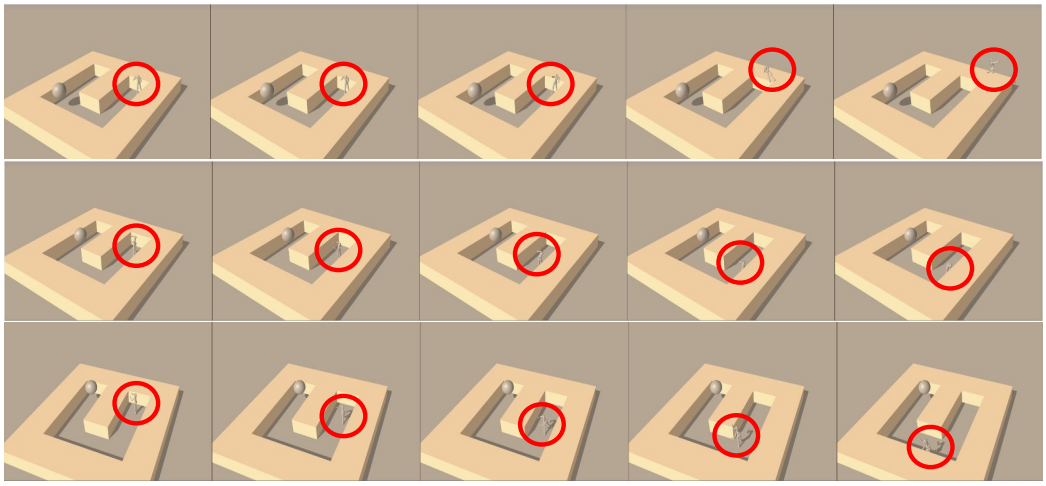

Figure 16: **Emergent Exploration Behavior in `humanoid-u-maze`.** C-TeC exhibits interesting emergent behaviors; for example, in the `humanoid-u-maze` environment, the agent learns to jump over the maze walls to escape the maze. Each row represents an independent evaluation epsidoe.

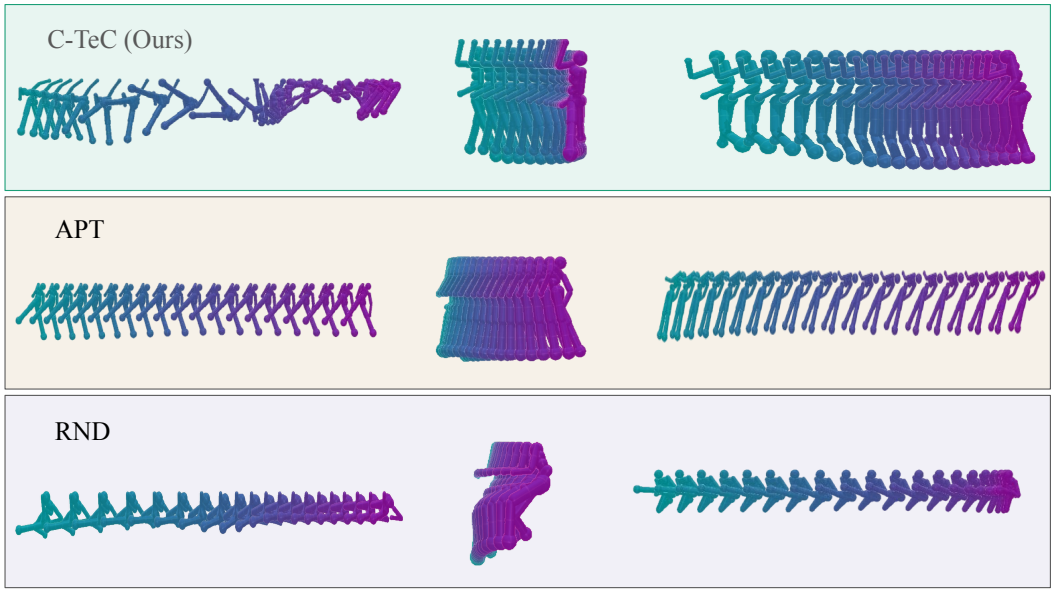

Figure 17: **Qualitative Comparison in `humanoid-u-maze`.**

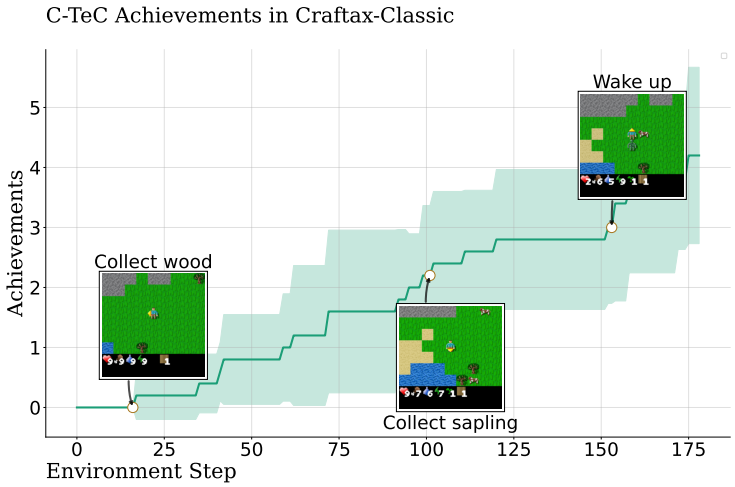

Figure 18: **C-TeC Achievements.** C-TeC unlocks interesting achievements in Craftax-Classic; the plot shows a subset of the unlocked achievements during an evaluation episode.