Constrained Intrinsic Motivation for Reinforcement Learning

Xiang Zheng[1] , **Xingjun Ma**[2] , **Chao Shen**[3] and **Cong Wang**[1*]
[1]City University of Hong Kong
[2]Fudan University
[3]Xi'an Jiaotong University
{xzheng235-c@my, congwang@}cityu.edu.hk, xingjunma@fudan.edu.cn, chaoshen@mail.xjtu.edu.cn

Abstract

This paper investigates two fundamental problems that arise when utilizing Intrinsic Motivation (IM) for reinforcement learning in Reward-Free Pre-Training (RFPT) tasks and Exploration with Intrinsic Motivation (EIM) tasks: 1) how to design an effective intrinsic objective in RFPT tasks, and 2) how to reduce the bias introduced by the intrinsic objective in EIM tasks. Existing IM methods suffer from static skills, limited state coverage, sample inefficiency in RFPT tasks, and suboptimality in EIM tasks. To tackle these problems, we propose *Constrained Intrinsic Motivation (CIM)* for RFPT and EIM tasks, respectively: 1) CIM for RFPT maximizes the lower bound of the conditional state entropy subject to an alignment constraint on the state encoder network for efficient dynamic and diverse skill discovery and state coverage maximization; 2) CIM for EIM leverages constrained policy optimization to adaptively adjust the coefficient of the intrinsic objective to mitigate the distraction from the intrinsic objective. In various MuJoCo robotics environments, we empirically show that CIM for RFPT greatly surpasses fifteen IM methods for unsupervised skill discovery in terms of skill diversity, state coverage, and fine-tuning performance. Additionally, we showcase the effectiveness of CIM for EIM in redeeming intrinsic rewards when task rewards are exposed from the beginning. Our code is available at https://github.com/x-zheng16/CIM.

1 Introduction

In the realm of Reinforcement Learning (RL), Intrinsic Motivation (IM) plays a vital role in the design of exploration strategies in both Reward-Free Pre-Training (RFPT) tasks and Exploration with Intrinsic Motivation (EIM) tasks [Barto, 2013]. It allows the RL agent to efficiently visit novel states by assigning higher intrinsic bonuses to unfamiliar states [Zhang *et al.*, 2021]. Current IM methods can be classified into three categories: knowledge-based, data-based, and competence-based IM methods [Laskin *et al.*, 2021].

Knowledge-based and data-based IM methods are employed in both RFPT and EIM tasks to encourage the agent to explore novel regions. Knowledge-based IM methods maximize the deviation of the agent's latest state visitation from the policy cover (i.e., the regions covered by all prior policies) [Zhang *et al.*, 2021]. These methods commonly estimate the density of the policy cover via the pseudo-count of state visit frequency [Bellemare *et al.*, 2016; Fu *et al.*, 2017], prediction errors of a neural network [Pathak *et al.*, 2017; Burda *et al.*, 2019], or variances of outputs of an ensemble of neural networks [Pathak *et al.*, 2019; Lee *et al.*, 2021; Bai *et al.*, 2021]. Data-based IM methods, on the other hand, directly maximize the state coverage (i.e., the region visited by the latest policy) via maximizing the state entropy [Hazan *et al.*, 2019; Mutti *et al.*, 2021; Liu and Abbeel, 2021b; Seo *et al.*, 2021]. However, knowledge-based and data-based IM methods are inefficient in RFPT tasks since they do not condition the latent skill variable, limiting the fine-tuning performance of the pre-trained policy in downstream tasks [Liu and Abbeel, 2021a]. Moreover, when utilized in EIM tasks, these IM methods introduce non-negligible biases to the policy optimization, leading to suboptimal policies [Chen *et al.*, 2022]. Specifically, intrinsic objectives may result in excessive exploration even when the task rewards are already accessible. This distraction induced by intrinsic objectives can deteriorate the performance of the RL agent and impede the wider application of these methods in EIM tasks.

Competence-based IM methods are designed for unsupervised skill discovery in RFPT tasks. They primarily maximize the mutual information between the state representation and the latent skill variable to learn a latent-conditioned poilcy [Gregor *et al.*, 2017; Sharma *et al.*, 2020; Laskin *et al.*, 2022]. The policy conditioned on the latent skill variable is required to change the state of the environment in a consistent and meaningful way, e.g., walking, flipping, pushing, to be finetuned efficiently in the downstream tasks. However, current competence-based IM methods have shown poor performance in the Unsupervised Reinforcement Learning Benchmark (URLB) [Laskin *et al.*, 2021], a benchmark of IM methods evaluated in RFPT tasks. Intuitively, directly maximizing the mutual information does not guarantee extensive state coverage or the discovery of dynamic skills and easily converges to simple and static skills due to the invariance of the mutual information to scaling and invert-

[*] Corresponding author.

ible transformation of the input variables [Park *et al.*, 2022; Park *et al.*, 2023]. Here, "dynamic" skills refer to skills that facilitate large state variations, e.g., running for locomotion tasks and moving for manipulation tasks. To address this limitation, Park et al. [2022] proposed Lipschitz-constrained Skill Discovery (LSD) to encourage dynamic skills. However, LSD suffers from severe sample inefficiency. The primary reason is that maximizing the intrinsic objective of LSD cannot guarantee maximum state entropy.

To overcome the limitations of existing knowledge-based, data-based, and competence-based IM methods, in this paper, we propose *Constrained Intrinsic Motivation (CIM)* which is 1) a novel constrained intrinsic objective in RFPT tasks, i.e., a lower bound of the conditional state entropy subject to an alignment constraint on the state encoder network, to make the RL agent discover dynamic and diverse (distinguishable) skills more efficiently; 2) a Lagrangian-based adaptive coefficient for the intrinsic objective in EIM tasks to alleviate the performance decrease due to the bias introduced by the intrinsic rewards.

In summary, we make the following main contributions:

- We propose *Constrained Intrinsic Motivation (CIM)* to overcome the limitations of knowledge/data-based and competence-based IM methods by combining the best of both worlds. CIM outperforms state-of-the-art IM methods, improving performance and sample efficiency in multiple MuJoCo robotics environments.
- CIM for RFPT introduces a lower bound for the state entropy, conditioning the state entropy on the latent skill variable without compromising the power of maximum state entropy exploration. CIM for RFPT also introduces a novel alignment constraint on the state encoder network. Compared with LSD, our CIM reduces the number of required samples by 20x less (e.g., from 400M to 20M in the environment Ant). Besides skill diversity and state coverage, our CIM achieves the highest fine-tuning performance in the Walker domain of URLB.
- CIM for EIM derives an adaptive coefficient of the intrinsic objective leveraging the constrained policy optimization method. We empirically demonstrate that the adaptive coefficient can effectively diminish the bias introduced by intrinsic bonuses in various MuJoCo tasks and improve the average task rewards.

2 Preliminaries

2.1 Markov Decision Processes

The discounted Markov Decision Process (MDP) is defined as $M = (\mathcal{S}, \mathcal{A}, P, R, \gamma, \mu)$, where $\mathcal{S}$ and $\mathcal{A}$ stand for the state space and the action space separately, $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition function mapping the state s and the action a to the distribution $P(s'|s, a)$ in the space of probability distribution $\Delta(\mathcal{S})$ over S, $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the reward function, $\gamma \in [0, 1)$ is the discount factor, and $\mu \in \Delta(\mathcal{S})$ is the initial state distribution. We focus on the episodic setting where the environment is reset once the agent reaches a final state s_f, a terminated state within the goal subsets $\mathbb{G}$ or a truncated state s_T. At the beginning of each episode, the agent samples a random initial state $s_0 \sim \mu$; at each time $t = 0, 1, 2, ..., T - 1$, it takes an action $a_t \in \mathcal{A}$ computed by a stochastic policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ or a deterministic one $\pi : \mathcal{S} \to \mathcal{A}$ according to the current state s_t and steps into the next state $s_{t+1} \sim P(\cdot|s_t, a_t)$ with an instant reward signal $r = R(s_t, a_t, s_{t+1})$ obtained.

2.2 Reward-Free Pre-Training and Exploration

RFPT and EIM are two types of intrinsically motivated RL tasks. To present the optimization objectives of RFPT and EIM, we first define the state distribution induced by the policy π as $d_\pi(s) = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t P(s_t = s|\mu, \pi) \in \mathcal{K}$, where $\mathcal{K}$ is the collection of all induced distributions. The extrinsic objective (the expectation of the task reward) is then $J_{\mathrm{E}}(d_\pi) = \mathbb{E}_{s\sim d_\pi} [r_{\mathrm{E}}]$, where $r_{\mathrm{E}} = R_{\mathrm{E}}(s, a, s')$ is the extrinsic task reward function. The intrinsic objective $J_{\mathrm{I}} : \mathcal{K} \to \mathbb{R}$ is defined as a differentiable function of the induced state distribution d_π with $L-$Lipschitz gradients.

In RFPT tasks, the task reward r^e is not available, and the agent aims to maximize only the intrinsic objective

$$L_k^{\mathrm{RFPT}}(\pi) = J_{\mathrm{I}}(d_\pi). \tag{1}$$

The agent can learn either a policy $\pi(a|s)$ without conditioning the latent skill variable when maximizing a knowledge-based or data-based intrinsic objective, or a latent-conditioned policy $\pi(a|s, z)$ when maximizing a competence-based intrinsic objective. Common evaluation metrics for RFPT tasks include state coverage, skill diversity, and fine-tuning performance in downstream tasks.

On the contrary, in EIM tasks, the goal of the agent is to complete a specific downstream task and maximize only the expected task rewards. The optimization objective of EIM is

$$L_k^{\mathrm{EIM}}(\pi) = J_{\mathrm{E}}(d_\pi) + \tau_k J_{\mathrm{I}}(d_\pi), \tag{2}$$

where τ_k is the coefficient of the intrinsic objective. Since the agent does not need to discover diverse skills for a specific task, $J_{\mathrm{I}}(d_\pi)$ in EIM tasks is commonly a knowledge-based or data-based intrinsic objective without conditioning the latent skill variable. The evaluation metric for EIM is only the expected task rewards.

2.3 Intrinsic Motivation Methods

We conduct a comprehensive comparison between our proposed CIM for RFPT and eighteen IM algorithms in regards of the intrinsic objective and the corresponding intrinsic reward function in Table 1, including Intrinic Curiosity Module (ICM) [Pathak *et al.*, 2017], Random Network Distillation (RND) [Burda *et al.*, 2019], Disagreement (Dis.) [Pathak *et al.*, 2019], MAximizing the DEviation from explored regions (MADE) [Zhang *et al.*, 2021], Adversarially Guided Actor-Critic (AGAC) [Flet-Berliac *et al.*, 2021], Maximum Entropy exploration (MaxEnt) [Hazan *et al.*, 2019], Active Pre-Training (APT) [Liu and Abbeel, 2021b], Random Encoders for Efficient Exploration (RE3) [Seo *et al.*, 2021], Variational Intrinsic Control (VIC) [Gregor *et al.*, 2017], Diversity Is All You Need (DIAYN) [Eysenbach *et al.*, 2019], Variational Intrinsic Successor featuRes (VISR) [Hansen *et al.*, 2020], Dynamics-Aware Discovery

Algorithm	**Intrinsic Objective**	**Intrinsic Reward**
ICM [Pathak *et al.*, 2017]	$\mathbb{E}_{\mathrm{s}}[\rho_\pi^{-1}(s)]$	$\hat{\rho}_\pi^{-1}(s)$
RND [Burda *et al.*, 2019]	$\mathbb{E}_{\mathrm{s}}[\rho_\pi^{-1}(s)]$	$\hat{\rho}_\pi^{-1}(s)$
Dis. [Pathak *et al.*, 2019]	$\mathbb{E}_{\mathrm{s}}[\rho_\pi^{-1}(s)]$	$\hat{\rho}_\pi^{-1}(s)$
MADE [Zhang *et al.*, 2021]	$\mathbb{E}_{\mathrm{s}}[(\rho_\pi^{-1}(s)d_\pi^{-1}(s))^{1/2}]$	$(\hat{\rho}_\pi^{-1}(s)\hat{d}_\pi^{-1}(s))^{1/2}$
AGAC [Flet-Berliac *et al.*, 2021]	$\mathbb{E}_{\mathrm{s}}[D_{\mathrm{KL}}(\pi(s) \mid \pi^\alpha(s))]$	$D_{\mathrm{KL}}(\pi(s) \mid \pi^\alpha(s))$
MaxEnt [Hazan *et al.*, 2019]	$H(\mathrm{s})$	$-\log \hat{d}_\pi(s)$
APT [Liu and Abbeel, 2021b]	$H(\mathrm{s})$	$-\log \hat{d}_\pi(f(s))$
RE3 [Seo *et al.*, 2021]	$H(\mathrm{s})$	$-\log \hat{d}_\pi(f(s))$
VIC [Gregor *et al.*, 2017]	$H(\mathrm{z}) - H(\mathrm{z} \mid \mathrm{s}_f)$	$\log q(z \mid s_f)$
DIAYN [Eysenbach *et al.*, 2019]	$H(\mathrm{z}) - H(\mathrm{z} \mid \mathrm{s}) + H(\mathrm{a} \mid \mathrm{s}, \mathrm{z})$	$\log q(z \mid s)$
VISR [Hansen *et al.*, 2020]	$H(\mathrm{z}) - H(\mathrm{z} \mid \mathrm{s})$	$S_c(\phi(s), z)$
DADS [Sharma *et al.*, 2020]	$H(\mathrm{s}' \mid \mathrm{s}) - H(\mathrm{s}' \mid \mathrm{s}, \mathrm{z})$	$-\log \hat{q}(s' \mid s) + \log q(s' \mid s, z)$
APS [Liu and Abbeel, 2021a]	$H(\phi(\mathrm{s})) - H(\phi(\mathrm{s}) \mid \mathrm{z})$	$-\log \hat{d}_\pi(\phi(s)) + S_c(\phi(s), z)$
CIC [Laskin *et al.*, 2022]	$H(\phi(\mathrm{s})),\ \text{s.t. } \phi \in \arg\min L^{\mathrm{CIC}}(\phi(s), z)$	$-\log \hat{d}_\pi(\phi(s))$
MOSS [Zhao *et al.*, 2022]	$\mathbb{E}_{\sim\mathcal{B}}(1-2m)H(\phi(\mathrm{s}) \mid m)$	$-(1-2m)\log \hat{d}_\pi(\phi(s))$
BeCL [Yang *et al.*, 2023]	$I(\mathrm{s};\mathrm{s}^+),\ \text{s.t. } \phi \in \arg\min L^{\mathrm{BeCL}}(\phi(s), z)$	$\exp(-l^{\mathrm{BeCL}})$
LSD [Park *et al.*, 2022]	$\mathbb{E}_{\mathrm{z,s}}(\phi(s') - \phi(s))^T z,\ \text{s.t. } \phi \in \arg\min L^{\mathrm{LSD}}(\phi(s), z)$	$(\phi(s') - \phi(s))^T z$
CSD [Park *et al.*, 2023]	$\mathbb{E}_{\mathrm{z,s}}(\phi(s') - \phi(s))^T z,\ \text{s.t. } \phi \in \arg\min L^{\mathrm{CSD}}(\phi(s), z)$	$(\phi(s') - \phi(s))^T z$
CIM (ours)	$H(\phi(\mathrm{s}) \mid \mathrm{z}),\quad \text{s.t. } \phi \in \arg\min L_a(\phi(s), z)$	$-\log \hat{d}_\pi(\phi(s)^T z \mid z)$

Table 1: A summarization of IM algorithms. We denote knowledge-based, data-based, and competence-based IM methods in red, green, and blue, respectively. 1) In knowledge-based IM methods (in red), ρ_π is the policy cover, $\hat{\rho}_p i$ is the estimated policy cover, d_π is the state distribution, $\hat{d}$ is the estimated state distribution, D_{KL} is the KL-divergence, and π^α is an adversarial policy in AGAC. 2) In data-based IM methods (in green), $H(\mathrm{s})$ stands for the entropy of the state distribution, and f stands for an image encoder in pixel-based tasks and an identity encoder when in state-based tasks. 3) In competence-based IM methods (in blue), z is the latent skill variable, q is the discriminator, $\hat{q}$ is the estimated probability density, s_f is the final state of the episode, S_c is the cosine similarity between two vectors, ϕ is the state encoder network, $s' \sim P(s'|s,a)$ is the subsequent state transitioned from the current state s when action a is taken, m is a Bernoulli variable, s^+ is the positive sample in the contrastive objective.

of Skills (DADS) [Sharma *et al.*, 2020], Active Pretraining with Successor features (APS) [Liu and Abbeel, 2021a], Contrastive Intrinsic Control (CIC) [Laskin *et al.*, 2022], Mixture Of SurpriseS (MOSS) [Zhao *et al.*, 2022], Behavior Contrastive Learning (BeCL) [Yang *et al.*, 2023], LSD, Controllability-Aware Skill Discovery (CSD) [Park *et al.*, 2023]. Among these methods, ICM, RND, Dis., MADE, and AGAC belong to knowledge-based IM methods since their intrinsic objectives depend on the agent's all historical experiences. MaxEnt, APT, and RE3 fall under data-based IM methods, directly maximizing the state entropy. Meanwhile, VIC, DIAYN, VISR, DADS, APS, CIC, MOSS, BeCL, LSD, and CSD are classified as competence-based methods, all of which condition the latent skill variable.

3 Constrained Intrinsic Motivation

In this section, we first present CIM for RFPT, a novel competence-based IM method that can learn dynamic and diverse skills efficiently. Specifically, we propose a constrained intrinsic objective $J_{\mathrm{I}}^{\mathrm{CIM}}$ for RFPT tasks, maximizing the conditional state entropy instead of the state entropy under a novel alignment constraint for the state representation. We then derive the corresponding intrinsic reward $r_{\mathrm{I}}^{\mathrm{CIM}}$ based on the Frank-Wolfe algorithm. Secondly, we propose CIM for EIM to adaptively adjust the coefficient of the intrinsic objective in Equation (2) based on constrained policy optimization to mitigate the bias introduced by the intrinsic objective. We then derive the adaptive coefficient τ_k^{CIM} based on the Lagrangian duality theory.

3.1 Constrained Intrinsic Motivation for RFPT

In this section, we develop CIM for RFPT, a novel constrained intrinsic objective for unsupervised RL. To better clarify the motivation for the design of the constrained intrinsic objective, we first review current coverage- and mutual information-based methods and analyze their limitations.

Problems of Previous Intrinsic Motivation Methods

Though knowledge-based and data-based IM methods may perform well in terms of state coverage in certain RFPT tasks, these methods lack awareness of latent skill variables and suffer from poor fine-tuning efficiency. To improve the fine-tuning performance in RFPT tasks, learning a latent-conditioned policy is necessary. However, existing competence-based IM methods perform poorly regarding skill diversity, state coverage, and sample inefficiency. We conjecture that there are two main issues:

Intrinsic objective. Maximizing only the mutual information is not suitable for dynamic and diverse skill discovery in RFPT tasks. Recall the two types of decomposition for the mutual information, that is, $I(\mathrm{s};\mathrm{z}) = H(\mathrm{z}) - H(\mathrm{z}|\mathrm{s}) = H(\mathrm{s}) - H(\mathrm{s}|\mathrm{z})$. Note that minimizing $H(\mathrm{z}|\mathrm{s})$ can be achieved with slight differences in states, and minimizing $H(\mathrm{s}|\mathrm{z})$ clearly impedes the maximization of $H(\mathrm{s})$. Thus, neither can encourage the agent to cover large state space. Moreover, using $H(\mathrm{s})$ directly as the intrinsic objective (e.g., CIC and MOSS) also leads to low state coverage, as shown in Figure 1. One of the key drawbacks of maximizing $H(\mathrm{s})$ is that it is challenging to estimate state density in the high-dimensional space.

Alignment constraint. Current alignment constraints for the state encoder network are not efficient enough. For instance, LSD can learn dynamic skills with Lipschitz constraint on the state encoder network but suffers from heavy sample inefficiency. On the other hand, CIC applies noise contrastive estimation to formulate the alignment constraint on the state encoder network but fails to learn sufficiently dynamic and diverse skills.

Design of Constrained Intrinsic Objective

To address the first issue, we propose choosing the conditional state entropy $H(\phi(\mathrm{s})|\mathrm{z})$ as the intrinsic objective. This is a key difference between our method and previous IM methods. Intuitively, by maximizing $H(\phi(\mathrm{s})|\mathrm{z})$, distances between adjacent states within the trajectories sampled by one skill are enlarged, which indicates a more *dynamic* skill.

For the second issue, we propose maximizing a novel lower bound of the mutual information between the state representation and the latent skill variable to make the trajectories sampled by different skills *distinguishable*, that is,

$$I(\phi(\mathrm{s});\mathrm{z}) \geq \log N - L_a(\phi(s), z), \tag{3}$$

where $L_a(\phi(s), z)$ is the alignment loss as follows:

$$\begin{aligned} L_a(\phi(s), z) &= \sum_i l_i^{\mathrm{CIM}}, \\ l_i^{\mathrm{CIM}} &= -\phi^{\mathrm{diff}}(\tau_i)^T z_i + \\ &\quad \log \sum\nolimits_{\tau_j \in S^- \bigcup \{\tau_i\}} \exp\left(\phi^{\mathrm{diff}}(\tau_j))^T z_i\right), \\ \phi^{\mathrm{diff}}(\tau) &= \phi(s') - \phi(s), \end{aligned} \tag{4}$$

N is the total number of samples for estimating the mutual information, $\tau = (s, s')$ is the slice of a trajectory, and S^- is a set of negative samples that contains trajectories sampled via skills other than z_i. We derive this lower bound based on Contrastive Predictive Coding [Oord *et al.*, 2018] by regarding the latent skill z as the context and $\phi^{\mathrm{diff}}(\tau)$ as the predictive coding. Based on Equation (3), the alignment constraint on the state encoder network $\phi(s)$ is

$$L_a(\phi(s), z) \leq C, \tag{5}$$

where C is a constant. Theoretically, as indicated in Table 1, C should represent the minimum of $L_a(\phi(s), z)$. In practice, we do not need to know the exact value of C. Instead, at each policy iteration step, we take several stochastic gradient descent steps on the alignment loss $L_a(\phi(s), z)$ to maximize the mutual information between the state representation $\phi(s)$ and the latent skill z.

The complete constrained intrinsic objective of CIM for RFPT is thus

$$\begin{aligned} &\max_{\pi} J_{\mathrm{I}}^{\mathrm{CIM}}(d_\pi(\phi(s))) = H(\phi(\mathrm{s})|\mathrm{z}) \\ &\text{s.t.} \quad L_a(\phi(s), z) \leq C, \end{aligned} \tag{6}$$

where $H(\phi(\mathrm{s})|\mathrm{z})$ is the conditional state entropy estimated in the state projection space, which depends on both the latent-conditioned policy network $\pi(\cdot|s, z)$ and the state encoder network $\phi(s)$, $d_\pi(\phi(s))$ is the distribution of the latent state $\phi(s)$ induced by the latent-conditioned policy $\pi(\cdot|s, z)$.

Interpreting $L_a(\phi(s), z)$ as an alignment loss provides us a novel insight to unify former competence-based IM methods. We can derive the alignment loss l_i of all previous competence-based IM methods listed in Table 1, e.g.,

$$\begin{aligned} l_i^{\mathrm{MSE}} &= \|\phi(s_i') - z_i\|_2^2, \\ l_i^{\mathrm{vMF}} &= -S_{\mathrm{c}}(\phi(s_i'), z_i), \\ l_i^{\mathrm{LSD}} &= -\phi^{\mathrm{diff}}(\tau_i)^T z_i + \lambda(\|\phi^{\mathrm{diff}}(\tau_i)\| - d(s, s')), \\ l_i^{\mathrm{CIC}} &= -S_{\mathrm{c}}\big(\phi(\tau_i), \phi_z(z_i)\big) + \\ &\quad \log \sum\nolimits_{\tau_j \in S^- \bigcup \{\tau_i\}} \exp\Big(S_{\mathrm{c}}\big(\phi(\tau_j), \phi_z(z_i)\big)\Big), \\ l_i^{\mathrm{BeCL}} &= -S_{\mathrm{c}}\big(\phi(s_i^+), \phi(s_i)\big) + \\ &\quad \log \sum\nolimits_{s_j \in S^- \bigcup \{s_i^+\}} \exp\Big(S_{\mathrm{c}}\big(\phi(s_j), \phi(s_i)\big)\Big), \end{aligned} \tag{7}$$

where $d(s, s')$ in l_i^{LSD} is the state distance function, S_{c} in l_i^{vMF} is the cosine similarity between two vectors, ϕ_z in l_i^{CIC} is a projection network for the latent skill vector, and $\phi(s_i^+)$ in l_i^{BeCL} is a state representation from a certain skill as the positive sample while all others are negative samples.

Estimation of Conditional State Entropy

We now introduce how to estimate the conditional state entropy $H(\phi(\mathrm{s})|\mathrm{z})$ involved in Equation (6). Recall the definition of the conditional state entropy

$$\begin{aligned} H(\phi(\mathrm{s})|\mathrm{z}) &= \mathbb{E}_{z \sim p_z}\left[H(\phi(\mathrm{s})|\mathrm{z} = z)\right] \\ &= \mathbb{E}_{z \sim p_z} \mathbb{E}_{\phi(s) \sim d_\pi}\left[-\log d_\pi(\phi(s))\right]. \end{aligned} \tag{8}$$

To estimate the outer expectation, we randomly sample the latent skill variables z from a prior distribution $p_z(z)$. For discrete skills, $p_z(z)$ can be a categorical distribution $\mathrm{Cat}(K, \boldsymbol{p})$ that is parameterized by $\boldsymbol{p}$ over a size-K the sample space, where p_i denotes the probability of the $i-$th skill. For continuous skills, we can select $p(z)$ as a uniform distribution $\mathcal{U}^{n_z}(a, b)$ over the interval $[a, b]$, where n_z is the dimension of the skill.

To estimate the inner expectation, we roll out trajectories using the latent-conditioned policy $\pi(\cdot|s, z)$ with z fixed. During the sampling phase, z is randomly sampled at the beginning of each episode and remains fixed throughout the entire trajectory. We store the state-skill pair (s,z) in the replay buffer. During the training phase, for each pair (s,z), we concatenate them as $[s, z]$ to be the input of the latent-conditioned policy $\pi(\cdot|s, z)$.

To estimate the state density d_π, instead of training a parameterized generative model, we leverage a more practical

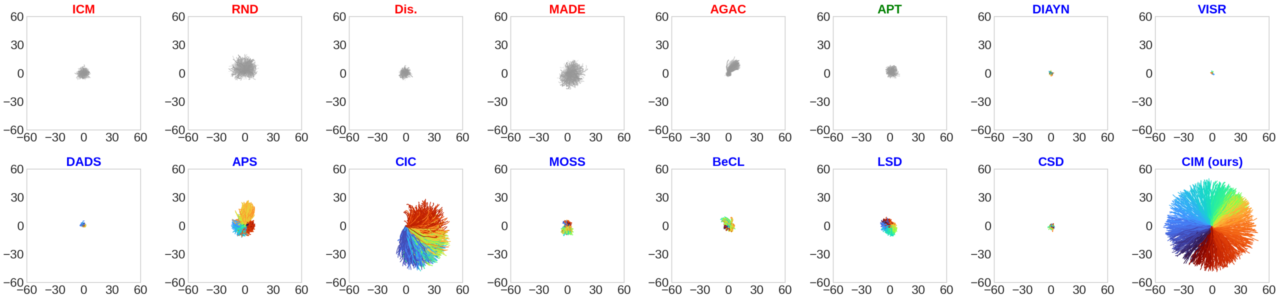


Figure 1: Visualization of 2D continuous locomotion skills in Ant. Each color of the trajectories in competence-based IM methods (in blue) represents the direction of the latent skill variable z.

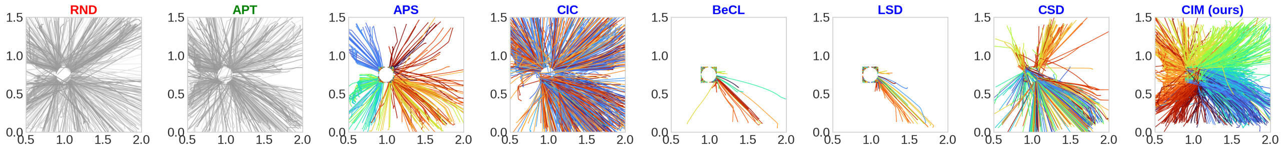


Figure 2: Visualization of 2D continuous manipulation skills discovered by various IM methods in FetchSlide. Each color of the trajectories in competence-based IM methods (in blue) represents the direction of the latent skill variable z.

non-parametric $\xi-$nearest neighbor ($\xi-$NN) estimator

$$\hat{d}_\pi(s_i) = \frac{1}{\lambda\left(B_\xi(s_i)\right)} \int_{B_\xi(s_i)} d_\pi(s)\mathrm{d}s, \tag{9}$$

where λ is the Lebesgue measure on $\mathbb{R}^d$, B_ξ is the smallest ball centered on s_i containing its ξ-th nearest neighbour s_i^ξ.

Lower bound of conditional state entropy. Each skill can be stochastic if we directly maximize the conditional state entropy $H(\mathrm{s}|\mathrm{z})$. To address this, we propose maximizing *the lower bound of* $H(\mathrm{s}|\mathrm{z})$ to encourage the skill z to produce large state variations along the direction of z in the latent space instead of being fully stochastic. To derive the lower bound of $H(\mathrm{s}|\mathrm{z})$, we first define a projection function $g_z(\phi(s)) = \phi(s)^T z$ for a fixed skill z. It is easy to verify that $H(\phi(\mathrm{s})|\mathrm{z}) \geq H(g_z(\phi(\mathrm{s}))|\mathrm{z}))$ with equality iff $S_c(\phi(s), z) = 1$, that is, $H(g_z(\phi(\mathrm{s}))|\mathrm{z}))$ is a lower bound of $H(\phi(\mathrm{s})|\mathrm{z})$. We thus can maximize $H(g_z(\phi(\mathrm{s}))|\mathrm{z}))$ to maximize $H(\phi(\mathrm{s})|\mathrm{z})$ and estimate the distribution of the *one-dimensional* random variable $g_z(\phi(s))$ for each z.

Intrinsic reward. Based on the above design, we can derive the intrinsic reward of CIM for RFPT as $r_{\mathrm{I}}^{\mathrm{CIM}}(s) = \log \|g_z(\phi(s)) - g_z(\phi(s))^\xi\|$. Here, $g_z(\phi(s))^\xi$ means the ξ-th nearest neighbor of $g_z(\phi(s))$. We adopted an average-distance version similar to APT to make training more stable:

$$r_{\mathrm{I}}^{\mathrm{CIM}}(s) = \log\left(1 + \frac{1}{\xi}\sum_{j=1}^{\xi} \|g_z(\phi(s)) - g_z(\phi(s))^j\|\right). \tag{10}$$

Intuitively, $r_{\mathrm{I}}^{\mathrm{CIM}}(s)$ measures how sparse the state s is in the projection subspace spanned by its corresponding latent skill z. This reward function can be justified based on the procedure of the Frank-Wolfe algorithm. Specifically, since L_k^{RFPT} is concave in d_π, maximizing L_k^{RFPT} involves solving $d_{\pi_{k+1}} \in \arg\max\langle \nabla_{d_\pi} L(d_{\pi_k}), d_{\pi_k} - d_\pi\rangle$ iteratively [Hazan *et al.*, 2019]. This iterative step is equivalent to policy optimization using a reward function proportional to $\nabla_{d_\pi} L(d_{\pi_k})$.

3.2 Constrained Intrinsic Motivation for EIM

In this section, we present our CIM for EIM, an adaptive coefficient for the intrinsic objective in Equation (2). Currently, IM methods for EIM tasks commonly use a constant coefficient or an exponentially decaying coefficient, which requires costly hyperparameter tunning. To avoid this, we propose reformulating Equation (2) by regarding the extrinsic objective as a constraint for the intrinsic objective, i.e.,

$$\max_{d_\pi \in \mathcal{K}} J_{\mathrm{I}}(d_\pi), \text{ s.t. } J_{\mathrm{E}}(d_\pi) \geq R_k, \tag{11}$$

where R_k represents the expected reward at the k-th iteration of policy optimization. We approximate R_k via $\hat{R}_k = \max_{j\in\{1,2,\ldots,k-1\}} J_{\mathrm{E}}(d_{\pi_j})$. We then leverage the Lagrangian method to solve Equation (11). The corresponding Lagrangian dual problem is $\min_{\lambda\geq 0}\max_{d_\pi} J_{\mathrm{I}}(d_\pi) + \lambda_k(J_{\mathrm{E}}(d_\pi) - \hat{R}_k)$. The Lagrangian multiplier λ is updated by Stochastic Gradient Descent (SGD), that is, $\lambda_k = \lambda_{k-1} - \eta(J_{\mathrm{E}}(d_{\pi_k}) - \hat{R}_{k-1})$ where η is the updating step size of λ_k. Observing that $\mathcal{L}_k(d_\pi, \lambda_k) \propto J_{\mathrm{E}}(d_\pi) + \lambda_k^{-1} J_{\mathrm{I}}(d_\pi)$, the adaptive coefficient τ_k^{CIM} is then derived as

$$\tau_k^{\mathrm{CIM}} = \min\{\{\lambda_{k-1} - \eta(J_{\mathrm{E}}(d_{\pi_k}) - \hat{R}_{k-1})\}^{-1}, 1\}, \tag{12}$$

where the outer minimization is to ensure numerical stability. As the Lagrangian multiplier λ_k grows, the penalty for the violation of τ_k^{CIM} gradually tends to zero; that is, the bias introduced by the intrinsic objective J_{I} is adaptively reduced.

Environment	RND	APT	APS	CIC	LSD	CSD	CIM (ours)
Ant (29D)	123 ± 15	33 ± 3	192 ± 75	697 ± 200	50 ± 24	4 ± 0	**1042 ± 158**
Humanoid (378D)	22 ± 1	22 ± 1	107 ± 33	64 ± 11	8 ± 1	4 ± 0	**1135 ± 360**
FetchPush (25D)	137 ± 22	**154** ± 17	79 ± 14	150 ± 34	24 ± 12	105 ± 48	141 ± 15
FetchSlide (25D)	182 ± 52	185 ± 49	178 ± 33	**223 ± 3**	31 ± 33	114 ± 79	187 ± 16

Table 2: State coverage of 2D continuous locomotion or manipulation skills discovered by various typical IM methods. We denote knowledge-based, data-based, and competence-based IM methods in red, green, and blue, respectively.

Task	DDPG	RND	Proto	APS	CIC	MOSS	BeCL	CIM (ours)
Flip	536 ± 66	470 ± 47	523 ± 89	407 ± 104	**709 ± 172**	425 ± 77	628 ± 46	664 ± 80
Run	274 ± 22	403 ± 105	347 ± 102	128 ± 38	492 ± 81	244 ± 13	467 ± 81	**585 ± 27**
Stand	931 ± 18	907 ± 16	861 ± 79	698 ± 215	939 ± 28	862 ± 100	**951 ± 3**	941 ± 21
Walk	777 ± 89	844 ± 99	828 ± 70	577 ± 133	905 ± 22	684 ± 40	781 ± 221	**921 ± 30**
Score	0.69 ± 0.23	0.72 ± 0.20	0.70 ± 0.20	0.49 ± 0.25	0.85 ± 0.18	0.60 ± 0.22	0.78 ± 0.19	**0.86 ± 0.11**

Table 3: Fine-tuning performance (average episode rewards ± standard deviations) of eight typical methods in Walker domain of URLB. We report the normalized average score in the last row. We denote knowledge-based, data-based, and competence-based IM methods in red, green, and blue, respectively.

Environment	l_i^{MSE}	l_i^{vMF}	l_i^{LSD}	l_i^{CIC}	l_i^{BeCL}	l_i^{CIM} (ours)
Ant (29D)	64	371	28	746	726	**1042**

Table 4: State coverage when replacing l_i^{CIM} in $L_a(\phi(s), z)$ with other alignment losses l_i as listed in Equation (7) in Ant.

Environment	$n_z = 2$	$n_z = 3$	$n_z = 10$	$n_z = 64$
Ant (29D)	**1042** ± 158	875 ± 240	901 ± 20	615 ± 54

Table 5: State coverage when varying the skill dimension n_z in Ant.

4 Experiments

4.1 Experimental Setup

Experimental setup for RFPT. We evaluate our intrinsic bonus $r_{\text{I}}^{\text{CIM}}$ for RFPT tasks on four Gymnasium environments, including two locomotion environments (Ant and Humanoid) and two manipulation environments (FetchPush and FetchSlide). We compare CIM for RFPT with fifteen IM methods in Table 1, including 1) four knowledge-based IM methods: ICM, RND, Dis., MADE, and AGAC; 2) one data-based IM method: APT; 3) and nine competence-based methods: DIAYN, VISR, DADS, APS, CIC, MOSS, BeCL, LSD, and CSD.

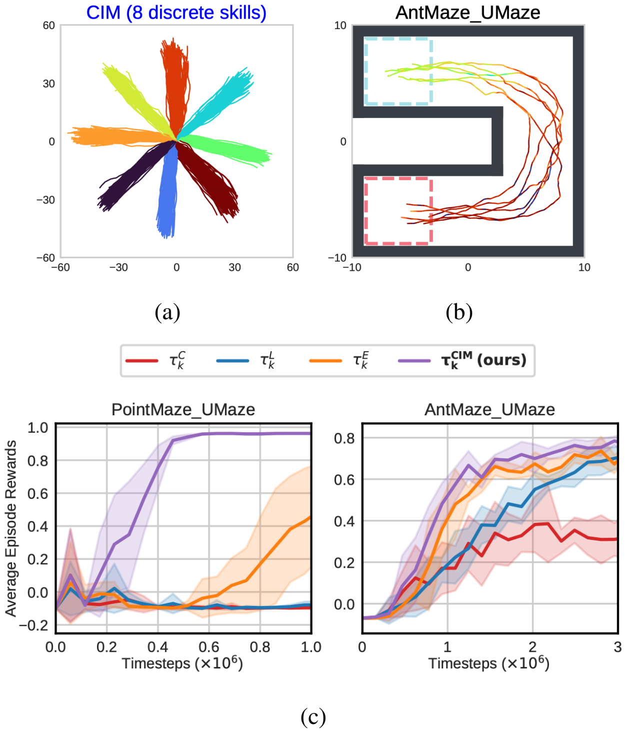
Experimental setup for EIM. We evaluate our adaptive coefficient τ_k^{CIM} for EIM in two navigation tasks (PointMaze_UMaze and AntMaze_UMaze) in D4RL [Fu *et al.*, 2020], and four sparse-reward tasks (SparseHalfCheetah, SparseAnt, SparseHumanoidStandup, and SparseGridWorld). τ_k^{CIM} is orthogonal with any intrinsic bonuses r_{I}. Unless otherwise mentioned, we adopt the state-of-the-art data-based intrinsic bonus $r_{\text{I}}^{\text{APT}} = \log(1 + {}^1\!/_k \sum_{j=1}^{k} \|\phi(s) - \phi(s)^j\|)$. The total instant reward is then $r = r_{\text{E}} + \tau_k^{\text{CIM}} r_{\text{I}}^{\text{APT}}$. We compare CIM for EIM with three baseline coefficient schemes, i.e., the constant coefficient $\tau_k^{\text{C}} \equiv 1$, the linearly decaying coefficient $\tau_k^{\text{L}} = (1 - k/T)$, and the exponentially decaying coefficient $\tau_k^{\text{E}} = 0.001^k$.

Coefficient	SHC	SA	SHS	SGW
τ_k^{C}	0.27	0.74	0.18	0.01
τ_k^{CIM} (ours)	**1**	**0.97**	**1**	**1**

Table 6: Test-time average episode rewards using different coefficient schemes across four sparse-reward EIM tasks.

4.2 Results in RFPT Tasks

Visualization of skills. As previous works like LSD do, we train CIM for RFPT to learn diverse locomotion continuous skills in the Ant and Humanoid environment and diverse manipulation skills in FetchPush and FetchSlide. The learned skills are visualized as trajectories of the agent on the $x - y$ plane in Figure 1 and Figure 2. Our CIM for RFPT outperforms all 15 baselines in terms of skill diversity and state coverage. The skills learned via CIM are interpretable because of our alignment loss; the direction of the trajectory on the $x - y$ plane changes consistently with the change in the direction of the skill. Specifically, CIM excels at learning dynamic skills that move far from the initial location in almost all possible directions, while most baseline methods fail to discover such diverse and dynamic primitives. Their trajectories are non-directional or less dynamic than CIM, especially in two locomotion tasks. Competence-based approaches like DIAYN, VISR, and DADS directly maximize the mutual information objectives but learn to take static postures instead of dynamic skills; such a phenomenon is also reported in LSD and CIC. Although APS and CIC can learn dynamic skills by directly maximizing the state entropy, CIM discovers skills that reach farther and are more interpretable via maximizing the lower bound of the state entropy. As for the two variants of CIC, MOSS and BeCL, they perform even worse than CIC in all tasks, reflecting their limitation in skill discovery. Lastly, LSD and CSD cannot learn dynamic skills within limited environment steps in Ant and Humanoid due to their low sample efficiency. Though they perform better in manipulation tasks than locomotion tasks, their learned skills are rambling compared with our CIM.



(a)

(b)

(c)

Figure 3: (a) Discrete CIM with $n_z = 8$ in Ant. (b) Trajectory visualization of the meta-controller where the color of each subtrajectory reflects the direction of the skill. (c) Learning curves using different coefficients of the intrinsic objective.

State coverage. To make a quantitative comparison between various IM methods, we measure their state coverage. The state coverage in Ant and Humanoid is determined by calculating the number of $2.5 \times 2.5\ \mathrm{m}^2$ bins occupied on the x-y plane, based on 1000 randomly sampled trajectories. This was then averaged over five runs. For FetchPush and FetchSlide, we use smaller bins. As shown in Table 2, CIM significantly outperforms all the baseline methods in two torque-as-input locomotion tasks and is comparable in two position-as-input manipulation tasks. Although the state coverage of CIM is slightly lower than APT and CIC in FetchPush and FetchSlide, the skills learned via CIM are more interpretable, as shown in Figure 2.

Fine-tuning efficiency in URLB. We also evaluate CIM for RFPT in URLB, a benchmark environment for RFPT in terms of fine-tuning efficiency. The results are presented in Table 3. The score (the last line of the table) is standardized by the performance of the expert DDPG, the same as in URLB and CIC. CIM performs better in Run and Walk tasks and achieves the highest average score. The dynamic skills learned through CIM for RFPT can be adapted quickly to diverse fine-tuning tasks, including flipping and standing. Our experiments also show that the skill dimension $n_z = 3$ is better for CIM to discover flipping skills than $n_z = 2$. The fixed skill selection mechanism for CIM is the same as CIC.

Ablation study. According to the results in Table 4, loss functions that follow the NCE style, such as l_i^{CIC}, l_i^{BeCL}, and l_i^{CIM}, perform better than other styles like MSE and vMF. Besides, l_i^{CIM} is the most effective. As shown in Figure 3a, our CIM can also be utilized to discover discrete diverse and dynamic skills, though it is mainly designed for continuous skills. Moreover, our CIM for RFPT is also robust to the number of skill dimensions, as shown in Table 5. Based on the ablation study, we can conclude that the two components of CIM, i.e., minimizing NCE-style alignment loss and maximizing conditional state entropy, are equally critical. Specifically, the results in Table 4 show that replacing the alignment loss of CIM with a trivial MSE loss reduces the state coverage in Ant from 1042 to 64. Moreover, Table 2 reveals that the state coverage achieved by CIM can reach 1135 in the challenging 378-dimensional Humanoid, while that achieved by CIC and BeCL, which use similar NCE-style alignment losses, is lower than 100.

4.3 Results in EIM Tasks

In PointMaze, we directly train a policy to control the Point without learning low skills since the environment dynamics are simple. In AntMaze, we train a meta-controller on top of the latent-conditioned policy pre-trained via our CIM for RFPT method. The meta-controller observes the target goal concatenated to the state observation $[s; s_g]$ and outputs the skill latent variable z at each timestep. We visualize the trajectories of the Ant in the $x - y$ plane as shown in Figure 3b, where the skills in a single trajectory gradually change to make the Ant turn a corner. Figure 3c shows that the Lagrangian-based adaptive coefficient τ_k^{CIM} outperforms three baseline coefficients, especially in PointMaze. Specifically, we can observe a small peak in the early stage of the training in PointMaze, which means the agent can reach the randomly generated target point with a small probability at the beginning. However, as the training processes, the agent is distracted by the intrinsic bonuses when using a trivial coefficient τ_k^{C} or τ_k^{L}. Moreover, other latent-conditioned policies are of poor quality, and we fail to train a mete-controller on top of those policies. We also conduct experiments to demonstrate the performance of CIM for EIM across four sparse-reward locomotion tasks. The results in Table 6 indicate that τ_k^{CIM} can effectively reduce the bias introduced by intrinsic rewards, thereby enhancing test-time average episode rewards in EIM tasks.

5 Conclusion

In this paper, we proposed Constrained Intrinsic Motivation (CIM) for RFPT and EIM tasks, respectively. For RFPT tasks, we designed a novel constrained intrinsic objective to discover dynamic and diverse skills. For EIM tasks, we designed an adaptive coefficient τ_k^{CIM} for the intrinsic objective based on constrained policy optimization. Our experiments demonstrated that CIM for RFPT outperformed all fifteen baselines across various MuJoCo environments regarding diversity, state coverage, sample efficiency, and fine-tuning performance. The latent-conditioned policy learned via CIM for RFPT was successfully applied to solve complex EIM tasks via training a meta-controller on top of it. We also empirically verified the effectiveness of our adaptive coefficient τ_k^{CIM} in multiple EIM tasks.

Acknowledgments

We thank the anonymous reviewers for their valuable feedback. This work was partially supported by HK RGC under Grants (CityU 11218322, R6021-20F, R1012-21, RFS2122-1S04, C2004-21G, C1029-22G, and N_CityU139/21) and the National Natural Science Foundation of China (U21B2018, 62161160337, 61822309, U20B2049, 61773310, U1736205, 61802166, 62276067).

References

[Bai *et al.*, 2021] Chenjia Bai, Lingxiao Wang, Lei Han, Jianye Hao, Animesh Garg, Peng Liu, and Zhaoran Wang. Principled exploration via optimistic bootstrapping and backward induction. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 577–587, 2021.

[Barto, 2013] Andrew G. Barto. Intrinsic motivation and reinforcement learning. In *Intrinsically Motivated Learning in Natural and Artificial Systems*, pages 17–47. 2013.

[Bellemare *et al.*, 2016] Marc G. Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Rémi Munos. Unifying count-based exploration and intrinsic motivation. In *Proc. of the Conference on Neural Information Processing Systems (NeurIPS)*, pages 1471–1479, 2016.

[Burda *et al.*, 2019] Yuri Burda, Harrison Edwards, Amos J. Storkey, and Oleg Klimov. Exploration by random network distillation. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2019.

[Chen *et al.*, 2022] Eric Chen, Zhang-Wei Hong, Joni Pajarinen, and Pulkit Agrawal. Redeeming intrinsic rewards via constrained optimization. In *Proc. of the Conference on Neural Information Processing Systems (NeurIPS)*, pages 4996–5008, 2022.

[Eysenbach *et al.*, 2019] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2019.

[Flet-Berliac *et al.*, 2021] Yannis Flet-Berliac, Johan Ferret, Olivier Pietquin, Philippe Preux, and Matthieu Geist. Adversarially guided actor-critic. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2021.

[Fu *et al.*, 2017] Justin Fu, John D. Co-Reyes, and Sergey Levine. EX2: Exploration with exemplar models for deep reinforcement learning. In *Proc. of the Conference on Neural Information Processing Systems (NeurIPS)*, pages 2577–2587, 2017.

[Fu *et al.*, 2020] Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4RL: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

[Gregor *et al.*, 2017] Karol Gregor, Danilo Jimenez Rezende, and Daan Wierstra. Variational intrinsic control. In *Proc. of the International Conference on Learning Representations (ICLR), Workshop Track*, 2017.

[Hansen *et al.*, 2020] Steven Hansen, Will Dabney, André Barreto, David Warde-Farley, Tom Van de Wiele, and Volodymyr Mnih. Fast task inference with variational intrinsic successor features. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2020.

[Hazan *et al.*, 2019] Elad Hazan, Sham M. Kakade, Karan Singh, and Abby Van Soest. Provably efficient maximum entropy exploration. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 2681–2691, 2019.

[Laskin *et al.*, 2021] 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 *Proc. of the Conference on Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2021.

[Laskin *et al.*, 2022] Michael Laskin, Hao Liu, Xue Bin Peng, Denis Yarats, Aravind Rajeswaran, and Pieter Abbeel. Unsupervised reinforcement learning with contrastive intrinsic control. In *Proc. of the Conference on Neural Information Processing Systems (NeurIPS)*, pages 34478–34491, 2022.

[Lee *et al.*, 2021] Kimin Lee, Michael Laskin, Aravind Srinivas, and Pieter Abbeel. SUNRISE: A simple unified framework for ensemble learning in deep reinforcement learning. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 6131–6141, 2021.

[Liu and Abbeel, 2021a] Hao Liu and Pieter Abbeel. APS: Active pretraining with successor features. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 6736–6747, 2021.

[Liu and Abbeel, 2021b] Hao Liu and Pieter Abbeel. Behavior from the void: Unsupervised active pre-training. In *Proc. of the Conference on Neural Information Processing Systems (NeurIPS)*, pages 18459–18473, 2021.

[Mutti *et al.*, 2021] Mirco Mutti, Lorenzo Pratissoli, and Marcello Restelli. Task-agnostic exploration via policy gradient of a non-parametric state entropy estimate. In *Proc. of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 9028–9036, 2021.

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

[Park *et al.*, 2022] Seohong Park, Jongwook Choi, Jaekyeom Kim, Honglak Lee, and Gunhee Kim. Lipschitz-constrained unsupervised skill discovery. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2022.

[Park *et al.*, 2023] Seohong Park, Kimin Lee, Youngwoon Lee, and Pieter Abbeel. Controllability-aware unsupervised skill discovery. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 27225–27245, 2023.

[Pathak *et al.*, 2017] Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 2778–2787, 2017.

[Pathak *et al.*, 2019] Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. Self-supervised exploration via disagreement. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 5062–5071, 2019.

[Seo *et al.*, 2021] Younggyo Seo, Lili Chen, Jinwoo Shin, Honglak Lee, Pieter Abbeel, and Kimin Lee. State entropy maximization with random encoders for efficient exploration. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 9443–9454, 2021.

[Sharma *et al.*, 2020] Archit Sharma, Shixiang Gu, Sergey Levine, Vikash Kumar, and Karol Hausman. Dynamics-aware unsupervised discovery of skills. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2020.

[Yang *et al.*, 2023] 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 *Proc. of the International Conference on Machine Learning (ICML)*, pages 39183–39204, 2023.

[Zhang *et al.*, 2021] Tianjun Zhang, Paria Rashidinejad, Jiantao Jiao, Yuandong Tian, Joseph E. Gonzalez, and Stuart Russell. MADE: Exploration via maximizing deviation from explored regions. In *Proc. of the Conference on Neural Information Processing Systems (NeurIPS)*, pages 9663–9680, 2021.

[Zhao *et al.*, 2022] Andrew Zhao, Matthieu Gaetan Lin, Yangguang Li, Yong-Jin Liu, and Gao Huang. A mixture of surprises for unsupervised reinforcement learning. In *Proc. of the Conference on Neural Information Processing Systems (NeurIPS)*, pages 26078–26090, 2022.