CURE: CONTRASTIVE PRETRAINING FOR ESCAPING SUBOPTIMAL ENTRAPMENT IN SKILL DISCOVERY

Anonymous authors
Paper under double-blind review

ABSTRACT

Unsupervised skill discovery enables agents to learn diverse and transferable behaviors without external rewards. Distance-maximizing skill discovery aligns representation spaces with state distances, which enables the agent to achieve strong exploration. However, its effectiveness in environments with irreversible failure modes (e.g., a humanoid robot falls and is unable to recover) is limited by the **suboptimal entrapment** issue: The coupled optimization of representation and policy causes the encoder to overfit to early-failure transitions, which assigns these transitions high intrinsic rewards, and thereby traps the policy in absorbing states. In this paper, we validate the suboptimal entrapment issue through theoretical analysis and empirical experiments. To address this issue, we propose CURE, a simple yet effective method that injects a robust geometric prior into the representation space. Before the joint optimization begins, CURE pretrains the state encoder on a minimal set of unlabeled trajectories using the InfoNCE contrastive objective. We prove that this pretraining is asymptotically equivalent to performing principal component analysis (PCA) on the transition dynamics, naturally favors forward-progressing transitions over failure ones, and successfully resolves the entrapment issue. Extensive experiments demonstrate that CURE significantly mitigates suboptimal entrapment, achieves broader state coverage, faster convergence, and robust behavioral priors for downstream tasks.

1 INTRODUCTION

Unsupervised learning has demonstrated strong effectiveness in natural language processing (Chen et al., 2020; Radford et al., 2021) and computer vision (Chen et al., 2020; Radford et al., 2021), where models learn meaningful representations from unlabeled data without task-specific supervision. As the reinforcement learning counterpart to this paradigm, *unsupervised skill discovery* enables agents to learn a diverse set of transferable behaviors without external rewards. Guided by intrinsic optimization objectives in reward-free environments, agents systematically explore state spaces, capture underlying transition dynamics, and obtain behavioral priors that facilitate adaptation to downstream tasks (Eysenbach et al., 2019).

Distance-maximizing skill discovery (DSD) (Park et al., 2022; 2023; 2024) has emerged as a leading class of methods for unsupervised skill discovery, achieving broad state coverage and strong performance. DSD learns a state representation space with a meaningful distance metric and trains agents to maximize traversal distance within this space. Despite its success in many domains, we identify an underexplored gap when DSD methods with static distance metrics operate in environments with irreversible failure modes, such as a humanoid robot falling (Gaspard et al., 2025; Falco et al., 2018). In these settings, agents can prematurely converge to absorbing states from which they cannot escape, failing to continue exploration or acquire diverse skills. As illustrated in Fig. 1, a high-DoF humanoid robot trained with standard DSD collapses and loses the ability to recover. We term this phenomenon the **suboptimal entrapment** issue, which is particularly crucial for real-world agents, for such failures can halt autonomous exploration and require costly manual reset.

In this work, we formalize the suboptimal entrapment issue and analyze its underlying mechanism. We find that the root cause of this issue is the coupling between representation learning and policy optimization. Specifically, during early exploration, the policy naturally encounters failure transitions, while the encoder overfits to these failure transitions to optimize the distance-maximizing

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

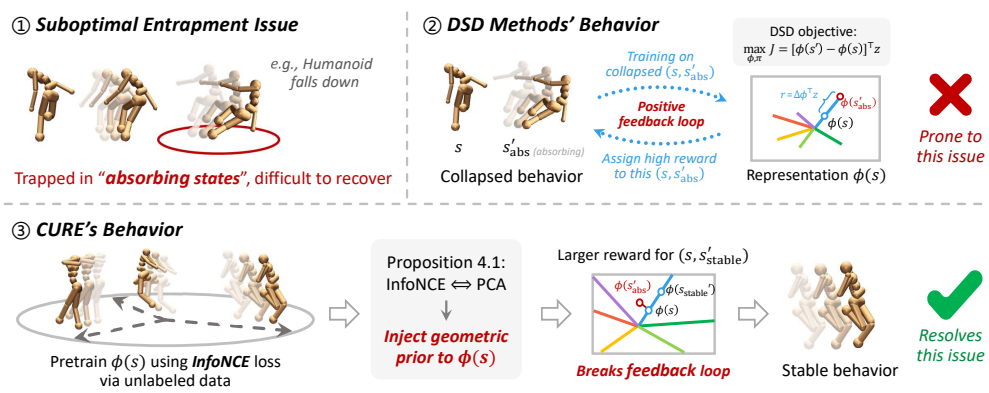


Figure 1: ① Illustration of the suboptimal entrapment issue. ② DSD methods are prone to this issue because of the positive feedback loop. ③ CURE breaks this feedback loop by pretraining $\phi(s)$.

objective, and assigns them high intrinsic rewards. This distorted reward further drives the policy into absorbing states, creating a self-reinforcing optimization loop. Taking METRA (Park et al., 2024) as a representative DSD method, we theoretically prove that the policy becomes trapped in these suboptimal states with high probability (Prop. 3.1), and experiments in Fig. 2 confirm that this issue severely limits both exploration depth and state coverage. We further prove that this collapse behavior constitutes a stationary point satisfying the Karush-Kuhn-Tucker (KKT) conditions, and escaping from it requires exponential time (Prop. 3.2 and 3.3).

To address this issue, we propose **C**ontrastive-**U**nsupervised **R**epres**E**ntation (CURE) for skill discovery, a simple yet effective method that injects a geometric prior into the representation space before the joint optimization process begins. CURE pretrains the state encoder ϕ on a minimal set of unlabeled, reward-free trajectories. It optimizes the encoder with a contrastive InfoNCE objective, which aligns transitions from the same trajectory and separates those from different ones. Then, the pretrained encoder is used to initialize standard skill training. Both theoretical and empirical results validate CURE's effectiveness. Theoretically, we prove that this pretraining is equivalent to performing temporal principal component analysis (PCA) (Hotelling, 1933) on the transition dynamics (Prop. 4.1). This equivalence provides a geometric prior that naturally assigns higher intrinsic reward to forward-progressing transitions than to failure ones, thereby preventing the reward distortion that causes entrapment (Cor. 4.1). Empirically, CURE significantly mitigates the suboptimal entrapment issue, achieves higher state coverage, and facilitates downstream tasks.

In summary, our contributions are threefold: (1) We identify the suboptimal entrapment issue, a critical failure mode for DSD methods with static distance metrics in environments with irreversible dynamics, and provide a complete theoretical analysis for the representative METRA algorithm. (2) We propose CURE, a simple yet theoretically grounded solution to the suboptimal entrapment issue, which uses contrastive pretraining to build a structured representation space. (3) Extensive experiments show that CURE consistently outperforms baselines, achieving state-space exploration, faster skill learning, and robust behavioral priors for downstream tasks.

2 Preliminaries

2.1 Unsupervised Skill Discovery

Unsupervised reinforcement learning typically considers a reward-free Markov decision process $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mu_0)$, where $\mathcal{S}$ and $\mathcal{A}$ are the state and action spaces, $\mathcal{P}$ the transition dynamics, and μ_0 the initial state distribution. Unsupervised skill discovery aims to acquire environment-specific knowledge by learning a set of distinguishable skills to explore the state space. This is achieved by optimizing a skill-conditioned policy $\pi(a|s, z)$ over a latent skill space $\mathcal{Z}$. Once learned, these skills can be leveraged as low-level primitives for downstream tasks.

A common approach to unsupervised skill discovery is to maximize the mutual information (MI) $I(s, z)$ between skills z and the visited states s. While early MI-based methods (Eysenbach et al., 2019) successfully discover distinguishable skills, they are prone to learning static behaviors. To

address this limitation, distance-maximizing skill discovery (DSD) methods have been developed, and we detail their formulation below.

2.2 Distance-maximizing Skill Discovery

As a prominent paradigm in unsupervised skill discovery, distance-maximizing skill discovery (DSD) (Park et al., 2023) jointly optimizes the state representation $\phi(s)$ and the skill-conditioned policy $\pi(a|s,z)$. Specifically, the distances in the representation space are enforced to align with distances in the raw state space, and the agent is trained to maximize displacement along skill z's direction within this representation space. Formally, the general DSD objective is:

$$\max_{\pi,\phi} J_{\text{DSD}} = \mathbb{E}_{(s,s')\sim\pi,z}(\phi(s') - \phi(s))^\top z, \quad \text{s.t.} \|\phi(x) - \phi(y)\|_2 \leq d(x,y), \quad \forall x, y \in \mathcal{S}, \tag{1}$$

where $d(x,y)$ is a specific distance metrics. METRA (Park et al., 2024), a representative DSD method, adopts temporal distance (Venkattaramanujam et al., 2019) as $d(x,y)$ because it is invariant to observation modality, and uses an equivalent constraint $\|\phi(s') - \phi(s)\|_2 \leq 1$ for all adjacent state pairs (s,s') within trajectories. METRA's objective is optimized via the Lagrange multiplier method, by updating ϕ to maximize Eq. 2, updating λ to minimize Eq. 3, and training π to maximize the accumulated intrinsic reward in Eq. 4:

$$J^\phi = \mathbb{E}_{(s,z,s')\sim\pi}[(\phi(s') - \phi(s))^\top z + \lambda \min(\delta, 1 - \|\phi(s') - \phi(s)\|_2^2)] \tag{2}$$

$$J^\lambda = \mathbb{E}_{(s,z,s')\sim\pi}[\lambda \min(\delta, 1 - \|\phi(s') - \phi(s)\|_2^2)] \tag{3}$$

$$r(s,z,s') = (\phi(s') - \phi(s))^\top z \tag{4}$$

where $\delta > 0$ denotes the slack parameter. Intuitively, this reward encourages moving far along the direction z to achieve broader state-space coverage and more dynamic behaviors than traditional MI-based methods (Park et al., 2022; 2024).

2.3 Absorbing Sets

In many complex tasks, agents encounter irreversible failure modes, e.g., a humanoid robot falling and being unable to stand up again (Gaspard et al., 2025; Falco et al., 2018). We formally characterize these failure modes as follows.

Definition 2.1 (Absorbing set)**.** *A non-trivial proper subset $\mathcal{M} \subsetneq \mathcal{S}$ is an absorbing set if $P(s' \in \mathcal{M} \mid s \in \mathcal{M}, a) = 1$ holds for all $s \in \mathcal{M}$ and $a \in \mathcal{A}$. An absorbing state is a special case where the absorbing set is a singleton.*

Once the agent enters an absorbing set, no further effective state space exploration occurs [1]. Consequently, the presence of absorbing sets creates a challenge for unsupervised skill discovery methods, which we analyze in Section 3.

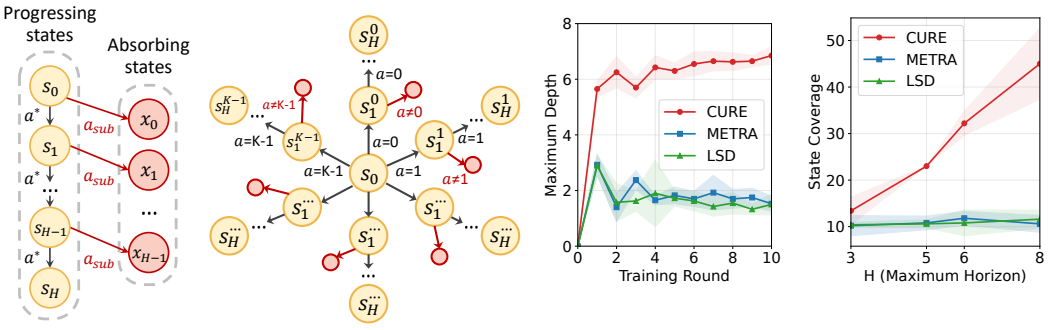
3 The Issue of Suboptimal Entrapment

In this section, we formalize the suboptimal entrapment issue: the skill-conditioned policy prematurely converges and traps the agent in an absorbing set, which prevents further exploration. Fig. 1 illustrates this failure with a humanoid agent that collapses and stops moving. Let $\Pi_{\mathcal{M}}$ denote the set of policies under which the agent eventually enters an absorbing set $\mathcal{M}$. Formally, the suboptimal entrapment issue occurs when a policy π maximizes the skill discovery objective $J(\pi, z)$ over Π_M, while remaining globally suboptimal:

$$\pi \in \arg\max_{\pi' \in \Pi_{\mathcal{M}}} J(\pi', z), \qquad J(\pi, z) < \max_{\pi'} J(\pi', z). \tag{5}$$

Because $\mathcal{M}$ is absorbing, the agent cannot escape to discover higher objective values, locking the policy into a suboptimal local maximum. To systematically study this issue, we first introduce an abstract MDP model to demonstrate how this failure mode occurs, and provide empirical evidence in Section 3.1. Then, we provide a theoretical explanation for this issue in Section 3.2.

[1] Although strictly irreversible states may be uncommon in idealized reversible simulations, certain failures can be effectively irreversible in practice. For unsupervised agents without task-specific rewards, recovery from states such as falling may have a very low probability within a practical interaction horizon. Moreover, real-world physical systems can exhibit genuinely irreversible failures, such as hardware damage caused by a fall. Such cases can be abstracted as absorbing sets, as discussed in Appendix F.1.



(a) (b) (c) (d)

Figure 2: (a) Illustrations of the Generalized Absorbing MDP. (b) Illustrations of its instantiation. (c) Maximum depth results and (d) state coverage results of CURE and DSD baselines.

3.1 Generalized Absorbing MDP and Empirical Evidence

To model environments with irreversible failures, we introduce the *Generalized Absorbing MDP* (Fig. 2(a)) as an analytical case. We construct this model as follows: **(1) State aggregation**: We partition the states into progressing states $s_0, \ldots, s_H$ and absorbing states $x_0, \ldots, x_{H-1}$. Let s_0 denote the set of initial states and H denote the episode horizon. For $h \geq 1$, s_h contains all states reachable from s_0 in exactly h steps without entering an absorbing set, and x_h is the union of all absorbing states reachable in a single step from s_h. **(2) Action aggregation**: We aggregate actions into an optimal action a^* and a suboptimal action a_{sub}. Action a^* keeps the agent in the progressing states. Action a_{sub} moves the agent into an absorbing state. **(3) Transitions**: Under this aggregation, transitions become deterministic: $P(s_{h+1} \mid s_h, a^*) = 1$ and $P(x_h \mid s_h, a_{\text{sub}}) = 1$ for every h, and each x_h is absorbing, i.e., $P(s' \in \mathcal{M} \mid x_h) = 1$. This abstraction captures a common structure in environments with irreversible failures and serves as a practical model for theoretical analysis.

To empirically illustrate the entrapment issue, we instantiate the Generalized Absorbing MDP, as shown in Fig. 2(b). The agent starts at the center, selects one of K directions, and must repeatedly select the same direction to move forward. Any other action permanently traps it in an absorbing state. We train DSD methods (METRA (Park et al., 2024) and LSD (Park et al., 2022)) and our method, CURE, on this task, with details in Appendix E.5. We evaluate two metrics: ① maximum depth, the furthest step reached along any direction before absorption or episode termination, and ② state coverage, the number of distinct states visited across all skills.

Fig. 2(c) shows that the maximum depth of DSD methods collapses after an initial peak, as the agent optimizes on absorbing transitions and thereby traps itself. We will formalize this mechanism in Prop. 3.1. In contrast, CURE steadily reaches deeper states, as its contrastively pretrained encoder (detailed in Section 4) captures forward progression and avoids absorption. Furthermore, Fig. 2(d) shows that DSD methods yield constant state coverage as the maximum horizon H increases, indicating convergence to suboptimal policies. Conversely, CURE achieves growing state coverage, proving that its pretrained representation enables broader exploration. These empirical observations validate the existence of the suboptimal entrapment issue, and motivate a formal analysis of why standard DSD methods fail to escape such traps, which we present next.

3.2 Theoretical Analysis of the Suboptimal Entrapment Issue

In this subsection, we theoretically analyze the mechanism of the suboptimal entrapment issue. We take the representative DSD method, METRA (Park et al., 2024), as an example, and analyze its training dynamics within the Generalized Absorbing MDP. We obtain the following proposition regarding early exploration dynamics, with proof provided in Appendix C.

Proposition 3.1. *Consider the Generalized Absorbing MDP defined above. Given an initial dataset $\mathcal{D}$ collected by a policy that selects a^* with probability $p_0 < 1$, for any integer $L < H$, the policy π_z optimized by METRA fails to reach s_{L+1} with probability at least $1 - Np_0^{L+1}$.*

Prop. 3.1 reveals the core mechanism of this failure. Without transitions to deeper progression states in the initial data, the encoder overfits to observed absorbing transitions and assigns them near-maximal intrinsic reward. This distorted reward landscape penalizes unexplored progressing transitions and eventually traps the policy at shallow progression states.

We further analyze the optimization dynamics of METRA after the policy enters these entrapments. We show that the optimization converges to a stationary point (Prop. 3.2), and the expected escape time grows exponentially with the episode length (Prop. 3.3), with proofs provided in Appendix C.

Proposition 3.2. *If the agent converges to a suboptimal behavior, the joint optimization of the policy π, representation ϕ, and Lagrange multiplier λ in METRA asymptotically approaches a KKT stationary configuration.*

Proposition 3.3. *Under stochastic gradient dynamics, the expected time to escape this suboptimal stationary point grows exponentially with the task episode length.*

Consequently, once the policy collapses into a failure behavior, the agent becomes trapped in a local optimum, effectively preventing the discovery of more advanced behaviors. To empirically verify this, we extend METRA training to 200k steps, but several skills remain trapped in the same failure behavior, with details provided in Appendix D.3.

While the above analysis focuses on METRA as a representative case, we provide extended analysis in Appendix C, showing that the entrapment mechanism generalizes across the broader DSD framework. These results collectively show that the suboptimal entrapment issue is a structural limitation of DSD methods in environments with irreversible failure modes. In Section 4, we address this issue by equipping the encoder with a geometry-aware initialization, which pre-aligns the representation space with progressing transitions, inherently preventing the policy from trapping in absorbing sets.

4 CURE: Contrastive Pretraining for Skill Discovery

The analysis in Section 3 identifies the suboptimal early-learned representation ϕ as the cause of suboptimal entrapment. We introduce CURE, a minimalist yet effective method that equips skill discovery with a geometric prior by pretraining the state encoder on a small set of unlabeled trajectories (Section 4.1). The pretrained encoder is then used to initialize standard skill learning. Theoretical analysis in Section 4.2 shows that this pretraining is equivalent to PCA on the unlabeled dataset, aligning the representation space with principal directions of environmental dynamics and mitigating the suboptimal entrapment issue. Fig. 1 and Algorithm 1 summarize the overall method.

4.1 Contrastive Pretraining for Representation

As established in Prop. 3.1 to 3.3, online exploration alone is insufficient to escape suboptimal traps in Generalized Absorbing MDP. We address this by pretraining the representation ϕ on a small unlabeled exploratory dataset $\mathcal{D}_{\text{exp}} = \{\tau_1, \ldots, \tau_N\}$ to introduce a geometric prior. Each trajectory $\tau_i = (s_0^i, \ldots, s_{H-1}^i)$ contains only action-free and reward-free state sequences. We optimize the encoder via the contrastive InfoNCE objective (Oord et al., 2018) over transition displacements $\Delta\phi = \phi(s') - \phi(s)$. Given a state transition (s, s') and its encoded displacement $\Delta\phi = \phi(s') - \phi(s)$, we form positive pairs $(\Delta\phi, \Delta\phi^+)$ by sampling another transition displacement $\Delta\phi^+$ from the same trajectory τ_i. Negative pairs $\Delta\phi_i^-$ are then sampled from transitions in different trajectories. Letting ω denote the parameters of encoder ϕ, the pretraining loss is:

$$\mathcal{L}_{\text{pre}}(\omega) = -\mathbb{E}_{\mathcal{D}_{\text{exp}}}\left[\log \frac{\exp(\Delta\phi^\top \Delta\phi^+/\tau)}{\exp(\Delta\phi^\top \Delta\phi^+/\tau) + \sum_{i=1}^{K} \exp(\Delta\phi^\top \Delta\phi_i^-/\tau)}\right]. \tag{6}$$

where $\tau > 0$ is the temperature hyperparameter and K is the number of negative pairs. Intuitively, this objective maps transitions from the same trajectory to similar latent directions, while pushing apart those from different trajectories. We also enforce a distance bound $\|\phi(s') - \phi(s)\|_2 \leq 1$ on all transitions in $\mathcal{D}_{\text{exp}}$, which stabilizes training and aligns with the Lipschitz condition in METRA.

After pretraining, we initialize the state encoder for skill discovery with these optimized parameters. We then update the encoder using Eq. 2 and train the policy $\pi(a|s, z)$ to maximize the intrinsic reward in Eq. 4. The full procedure is summarized in Algorithm 1.

4.2 Theoretical Analysis

In this subsection, we show that our contrastive pretraining approach is not merely intuitive but theoretically grounded. Specifically, optimizing the InfoNCE loss is mathematically equivalent to performing principal component analysis (PCA) (Hotelling, 1933) on the unlabeled dataset. We formalize this equivalence in Proposition 4.1 (proof in Appendix C).

Proposition 4.1. *Assume the encoder $\phi(s)$ is a linear projection of a temporally consistent embedding $\psi(s)$, where ψ satisfies $d_{temp}(u, v) = \|\psi(u) - \psi(v)\|_2$ for all $u, v \in \mathcal{S}$, $d_{temp}(u, v) < \infty$, with $d_{temp}(u, v)$ denoting the temporal distance (Venkattaramanujam et al., 2019), and transition pairs from the same trajectory exhibit consistent displacement directions. Then, optimizing the contrastive InfoNCE loss in Eq. 6 over $\mathcal{D}_{exp}$ is asymptotically equivalent to performing PCA on the covariance matrix $\Sigma = \mathbb{E}_{(s,s')\sim\mathcal{D}_{exp}}[(\psi(s') - \psi(s))(\psi(s') - \psi(s))^\top]$.*

This equivalence shows that the pretrained encoder ϕ extracts the dominant directions of state displacement in the dataset. If the exploratory dataset contains a non-negligible proportion of progressing transitions, the sustained and coherent displacements produced by progressing trajectories, such as steady walking in humanoid control, make their dynamics more likely to be captured by the leading principal components than directions associated with failure transitions.

Furthermore, we demonstrate that this pretraining provides METRA with a structured initialization. By aligning the representation space with the principal directions of environmental dynamics, the intrinsic reward landscape for subsequent policy optimization is modified: the intrinsic reward better supports forward state progression and steers the policy away from absorbing states. We formalize this property in Cor. 4.1 (proof in Appendix C).

Corollary 4.1. *Under the Generalized Absorbing MDP setup of Prop. 3.1, let s_L denote the exploration frontier in the initial dataset. Assuming that the progressing displacement is sufficiently aligned with the first principal component. Given an encoder pretrained on trajectories that extend beyond s_L to depth $L + 1$, and a skill z aligned with the first principal component, the value of the optimal action a^* at s_L satisfies $Q(s_L, a^*, z) \geq Q(s_L, a_{sub}, z) - \delta'$ for a small constant δ'.*

Cor. 4.1 explains how CURE mitigates the suboptimal entrapment issue analyzed in Prop. 3.1. Without pretraining, unvisited progressing transitions yield near-zero reward, which steers the policy to prefer observed failure paths. In contrast, the pretrained encoder aligns displacements in the representation space with dominant motion directions, allowing unexplored progressions to receive meaningful intrinsic rewards. Thus, this structural prior reduces the Q-value gap that traps the policy, making it more likely to escape the entrapment and sustain exploration.

We would like to note that our theory imposes a mild requirement on the exploratory dataset: the exploratory dataset contains a basic level of progressive transitions, such as (s_L, s_{L+1}), to capture the relevant variance, without requiring expert demonstrations or globally optimal trajectories. As confirmed by the empirical ablations in Section 5.3, CURE remains robust when the dataset contains considerable failure trajectories, is heavily truncated, or covers only a small region of the state space.

4.3 Implementation Details

Exploration-triggered adaptive learning rate mechanism. During early skill discovery, a fixed learning rate allows suboptimal updates to quickly overwrite the pretrained prior, which destabilizes optimization. We therefore adjust the learning rate based on the agent's exploration progress. Specifically, under the Lipschitz constraint $\|\phi(s_{t+1}) - \phi(s_t)\|_2 \leq 1$, the expected norm of the terminal state embedding $\mathbb{E}[\|\phi(s_{H-1}) - \phi(s_0)\|_2]$ serves as a practical proxy for state-space coverage. We set the learning rate α as:

$$\alpha = \begin{cases} \alpha_{\text{low}}, & \text{if } \mathbb{E}[\|\phi(s_{H-1}) - \phi(s_0)\|_2] < \eta H, \\ \alpha_{\text{normal}}, & \text{otherwise}, \end{cases} \tag{7}$$

with $\eta = 0.5$. This design applies a low learning rate to preserve the pretrained structure when exploration is limited, and resumes the normal rate for fine-tuning when the agent reliably traverses the environment. We empirically validate this technique via ablation studies in Section 5.3.

Overall algorithm. Algorithm 1 and Fig. 1 illustrate the procedure of CURE. We first pretrain the encoder ϕ on the exploratory dataset $\mathcal{D}_{\text{exp}}$ using the InfoNCE objective in Eq. 6 while enforcing

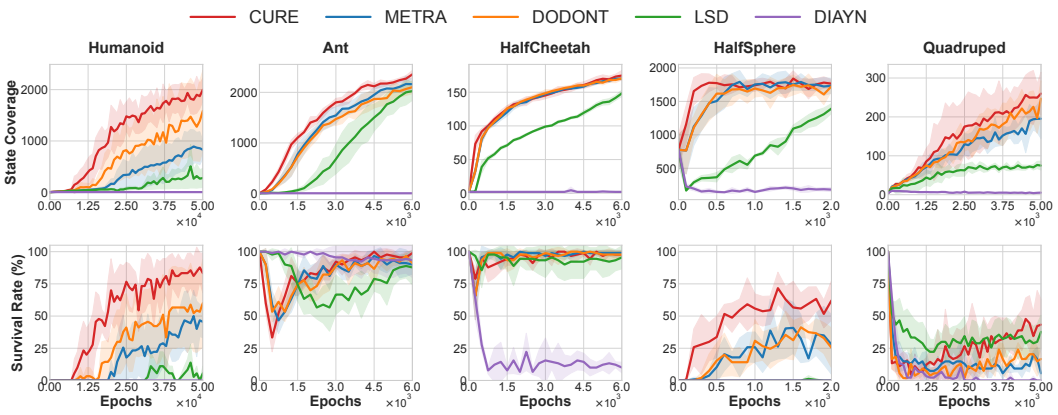


Figure 3: Comparison of state coverage (top) and survival rate (bottom) over training epochs. Solid lines and shaded regions represent the mean and standard deviation across five seeds. CURE consistently outperforms baselines in both exploration efficiency and behavioral stability.

the constraint via a dual method (lines 2∼9). During the subsequent skill discovery process (lines 10∼14), the pretrained encoder is used as initialization, and the encoder learning rate is adaptively controlled as described above.

5 Experiment

We conduct extensive experiments to answer the following questions: *Q1*: Does CURE effectively mitigate the suboptimal entrapment issue and improve the efficiency of discovering diverse and meaningful skills? *Q2*: Does the contrastive pretraining effectively capture the principal components of transition dynamics and construct a geometrically well-structured representation space? *Q3*: What is the individual contribution of each proposed technique in CURE?

5.1 Setup

Domains. We evaluate CURE across complex robotic locomotion domains, including Humanoid, Ant, HalfCheetah from MuJoCo (Todorov et al., 2012; Brockman et al., 2016), and the pixel-based Quadruped from the DeepMind Control Suite (Tassa et al., 2018). We use colored floors in the Quadruped environment to enable the agent to infer its location from pixels (Park et al., 2024). We configure these environments to terminate episodes upon unrecoverable physical failures, such as falling or flipping over. This setup aligns with real-world robotic constraints, where such events cause a loss of control and necessitate expensive manual resets (Rudin et al., 2022). Complementing these high-dimensional tasks, we also design a custom environment, HalfSphere, to isolate the suboptimal entrapment issue. As illustrated in Fig. 4, a mass point initialized at the top of an inverted hemisphere must navigate downward under gravity. The agent controls the mass point by applying tangential forces along the hemisphere's surface. The agent naturally slides down, but excessive acceleration causes it to fall off the surface into an absorbing state, forcing it to balance active exploration against the risk of terminal failures. Details of environment configurations are provided in Appendix E.1.

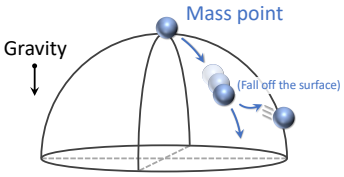


Figure 4: HalfSphere.

Baselines and implementation. We compare CURE with two primary groups of baselines. (1) *Classic unsupervised skill discovery*: This group evaluates purely intrinsic exploration, including a mutual information-based method, DIAYN (Eysenbach et al., 2019), and two distance-maximizing skill discovery (DSD) methods, METRA (Park et al., 2024) and LSD (Park et al., 2022). (2) *Data-guided skill discovery*: This category leverages unlabeled datasets to inject prior knowledge into the skill discovery process. Within this group, we evaluate DoDont (Kim et al., 2024), which trains an instruction network to identify desirable behaviors and dynamically reweights the intrinsic reward using this network. For the pretraining phase of CURE and the data-guided baseline, we utilize

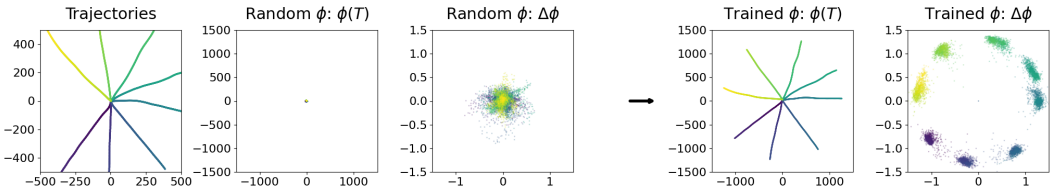


Figure 5: (a) Visualizations of trajectories of Humanoid in x-y space. (b, d) Trajectories in ϕ-space before and after CURE's pretraining. (c, e) Transition displacements $\Delta\phi$ of these trajectories before and after CURE's pretraining.

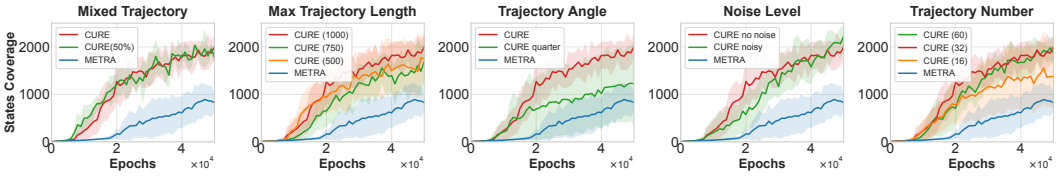


Figure 6: State coverage results of CURE under various data qualities in the MuJoCo Humanoid environment. We modify (a) the proportion of failed transitions, (b) the trajectory length, (c) the range of coverage angle, (d) observation noise, and (e) the number of trajectories.

32 and 64 unlabeled trajectories as the exploratory dataset for state-based and pixel-based tasks. We set the InfoNCE temperature to $\tau = 0.2$ and employ the adaptive learning rate mechanism in Section 4.3, with $\alpha_{\text{normal}}, \alpha_{\text{low}}$ are $6 \times 10^{-4}, 3 \times 10^{-4}$ for Humanoid and $10^{-4}, 2 \times 10^{-5}$ for other environments. More details are provided in Appendix E.4.

Evaluation metrics. To quantitatively evaluate the performance of CURE, we employ three metrics: (1) *State coverage* measures the agent's ability to explore the state space. Following (Park et al., 2024), this metric is computed by counting the unique 1×1 x-y bins (or 1-unit x-axis bins for HalfCheetah, $1\times1\times1$ x-y-z bins for HalfSphere) visited by the agent across all evaluation trajectories. (2) *Survival rate* quantifies the agent's robustness against the suboptimal entrapment issue, calculated as the fraction of episodes that reach 75% (Humanoid) or 100% (other environments) of the maximum episode horizon without early termination. (3) *Downstream task performance* evaluates skill utility by reporting the cumulative reward of a high-level controller, which uses the learned skills as frozen low-level primitives. More details of these metrics are provided in Appendix E.2.

5.2 Main Results

Enhanced state coverage and skill diversity. As shown in Fig. 3, CURE consistently achieves higher state coverage and exploration efficiency than baselines. This advantage is particularly evident in Humanoid and HalfSphere, where agents are highly susceptible to early termination and absorbing states. In Humanoid, while methods like METRA and LSD struggle to explore more broadly due to the suboptimal entrapment issue, CURE maintains steady growth in state coverage. Qualitative visualizations in Appendix D further confirm that CURE learns diverse and widely distributed locomotion patterns.

Mitigating the entrapment issue. To verify that CURE successfully addresses the suboptimal entrapment issue, we analyze the survival rate of agents during training. As shown in Fig. 3, CURE achieves a higher survival rate than baselines across environments. We find that DSD methods like LSD and METRA frequently suffer early terminations due to falls or flips, which lead to absorbing states. By injecting a geometric prior into the encoder, CURE substantially reduces the frequency of such failures. This demonstrates that CURE encourages the agent to learn sustainable locomotive behaviors rather than collapsing to suboptimal behaviors that lead to absorbing states.

Visualization of the representation space. To verify the theoretical claim that our contrastive objective effectively performs PCA on trajectory dynamics, we visualize trajectories in $\mathcal{D}_{\text{exp}}$ in the learned ϕ-space. Fig. 5 shows that the initial distribution of $\Delta\phi = \phi(s') - \phi(s)$ lacks a clear structure, while after pretraining, the scatter plots of $\Delta\phi$ reveal an axis-aligned structure. This ob-

servation is consistent with Prop. 4.1, that the pretrained encoder captures the principal components of state transitions. Furthermore, the trajectory layout in the ϕ-space aligns with its layout in the original state space, confirming that the representation preserves meaningful physical dynamics.

We further evaluate the utility of discovered skills on downstream zero-shot and hierarchical control tasks in Appendix D.1, where CURE produces more capable and transferable skills.

5.3 Ablation Studies

Robustness to dataset quality. We investigate the sensitivity of CURE to the quality and diversity of the exploratory dataset. The results are shown in Fig. 6.

(1) *Failure-mixed trajectories:* Increasing the fraction of failed trajectories in $\mathcal{D}_{\text{exp}}$ to 50% causes almost no drop in CURE's performance, showing CURE's robustness to a high fraction of failure data. Additional results on HalfSphere are provided in Appendix D.2.

(2) *Trajectory length:* We truncate trajectories (original 1000 steps) to 750 and 500 steps. The final state coverage decreases marginally and remains significantly higher than baselines, showing robustness to trajectory length.

(3) *Coverage angle:* We restrict the dataset to trajectories that cover only one quarter of the standard x-y area. The state coverage remains above METRA, showing CURE's ability to extract a meaningful prior even from partial data. However, the declined performance and increased variance indicate that reduced coverage makes training less stable.

(4) *Observation noise:* Injecting Gaussian noise with $\sigma = 0.01$ into the trajectory states only marginally affects the training dynamics and final state coverage. This confirms that our contrastive objective extracts robust motion patterns without overfitting to local perturbations.

(5) *Number of trajectories:* We vary the number of trajectories among 16, 32, and 60. Performance with 16 trajectories shows a moderate drop but still surpasses baselines, while increasing to 60 trajectories yields no significant gains over 32. These confirm CURE's sample efficiency and robustness to dataset size.

Moreover, we provide additional ablation results in Appendix D.2, which demonstrate CURE's robustness to the InfoNCE temperature and the necessity of the adaptive learning rate.

6 Related Work

Early skill discovery methods maximize mutual information between latent skills and visited states (Eysenbach et al., 2019; Sharma et al., 2020; Liu & Abbeel, 2021; Laskin et al., 2022). Distance-maximizing methods encourage broader exploration by learning state representations with meaningful distance metrics (Park et al., 2022; 2023; 2024), but can struggle in environments with irreversible failure modes, as analyzed in Section 3. Recent methods incorporate prior knowledge into skill discovery through data-driven guidance (Kim et al., 2024), analytical constraints (Kim et al., 2023), or human preferences (Hussonnois et al., 2023; 2025). Our work follows this direction by introducing a geometric prior from unlabeled trajectories to mitigate the suboptimal entrapment issue. Within skill discovery, (Laskin et al., 2022) optimize a contrastive objective on online data to maximize mutual information. In contrast, CURE uses contrastive learning as a pretraining stage to establish a geometrically meaningful representation, which mitigates the suboptimal entrapment issue. A more detailed discussion is provided in Appendix B.

7 Conclusion

In this paper, we identify and theoretically analyze the suboptimal entrapment issue, a critical failure mode for distance-maximizing skill discovery in environments with irreversible dynamics. We propose CURE to resolve this issue, which pretrains a state encoder on minimal unlabeled trajectories via contrastive learning. This pretraining is proven to be equivalent to performing PCA on transition dynamics, providing a geometric prior that suppresses the failure modes. Experiments show that CURE effectively resolves the entrapment issue, achieves higher state coverage, and produces robust behavioral priors for downstream tasks. Our work presents a lightweight yet theoretically grounded approach to inject geometric priors into unsupervised reinforcement learning.

AI USE STATEMENT

In this work, we used generative AI tools for language editing and manuscript polishing, including improving grammar, clarity, and concision. We have not used generative AI tools for method or theory development, formulation or proof of mathematical claims, hypothesis development, research methodology or experiment design, method implementation, translation, dataset processing, qualitative analysis, or result interpretation. Synthetic data generation is not applicable to this work. We have reviewed all AI-assisted work. AI-assisted text was manually checked by the authors for technical correctness and consistency with the intended meaning. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide complete proofs for the theoretical results and detailed experimental settings in the appendix. The appendix also includes implementation details, hyperparameters, and environment configurations used in our experiments. The source code used to produce the experimental results is provided in the supplementary materials.

REFERENCES

Rajendra Bhatia. *Matrix Analysis*, volume 169 of *Graduate Texts in Mathematics*. Springer Science & Business Media, 2013. ISBN 978-1-4614-7139-5. doi: 10.1007/978-1-4614-7139-5.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

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.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15750–15758, 2021.

Benjamin Eysenbach, Julian Ibarz, Abhishek Gupta, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *7th International Conference on Learning Representations, ICLR 2019*, 2019.

Benjamin Eysenbach, Tianjun Zhang, Sergey Levine, and Russ R Salakhutdinov. Contrastive learning as goal-conditioned reinforcement learning. *Advances in Neural Information Processing Systems*, 35:35603–35620, 2022.

Pietro Falco, Abdallah Attawia, Matteo Saveriano, and Dongheui Lee. On policy learning robust to irreversible events: An application to robotic in-hand manipulation. *IEEE Robotics and Automation Letters*, 3(3):1482–1489, 2018.

Mark Iosifovich Freidlin and Alexander D Wentzell. Random perturbations. In *Random perturbations of dynamical systems*, pp. 15–43. Springer, 1998.

Clément Gaspard, Marc Duclusaud, Grégoire Passault, Mélodie Daniel, and Olivier Ly. Frasa: An end-to-end reinforcement learning agent for fall recovery and stand up of humanoid robots. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 15994–16000. IEEE, 2025.

Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417, 1933.

Maxence Hussonnois, Thommen George Karimpanal, and Santu Rana. Controlled diversity with preference: Towards learning a diverse set of desired skills. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, pp. 1135–1143, 2023.

Maxence Hussonnois, Thommen George Karimpanal, and Santu Rana. Human-aligned skill discovery: Balancing behaviour exploration and alignment. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, pp. 1025–1033, 2025.

Hyunseung Kim, BYUNG KUN LEE, Hojoon Lee, Dongyoon Hwang, Donghu Kim, and Jaegul Choo. Do's and don'ts: Learning desirable skills with instruction videos. *Advances in Neural Information Processing Systems*, 37:47741–47766, 2024.

Sunin Kim, Jaewoon Kwon, Taeyoon Lee, Younghyo Park, and Julien Perez. Safety-aware unsupervised skill discovery. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 894–900. IEEE, 2023.

Hendrik Anthony Kramers. Brownian motion in a field of force and the diffusion model of chemical reactions. *physica*, 7(4):284–304, 1940.

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, Hao Liu, Xue Bin Peng, Denis Yarats, Aravind Rajeswaran, and Pieter Abbeel. Unsupervised reinforcement learning with contrastive intrinsic control. *Advances in Neural Information Processing Systems*, 35:34478–34491, 2022.

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.

Hao Liu and Pieter Abbeel. Aps: Active pretraining with successor features. In *International Conference on Machine Learning*, pp. 6736–6747. PMLR, 2021.

Marlos C Machado, Marc G Bellemare, and Michael Bowling. A laplacian framework for option discovery in reinforcement learning. In *International Conference on Machine Learning*, pp. 2295–2304. PMLR, 2017.

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

Seohong Park, Jongwook Choi, Jaekyeom Kim, Honglak Lee, and Gunhee Kim. Lipschitz-constrained unsupervised skill discovery. In *International Conference on Learning Representations*, 2022.

Seohong Park, Kimin Lee, Youngwoon Lee, and Pieter Abbeel. Controllability-aware unsupervised skill discovery. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 27225–27245, 2023.

Seohong Park, Oleh Rybkin, and Sergey Levine. Metra: Scalable unsupervised rl with metric-aware abstraction. In *International Conference on Learning Representations*, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Gareth O Roberts and Richard L Tweedie. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, 2(3):341–363, 1996.

Nikita Rudin, David Hoeller, Philipp Reist, and Marco Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In *Conference on Robot Learning*, pp. 91–100. PMLR, 2022.

Archit Sharma, Shixiang Gu, Sergey Levine, Vikash Kumar, and Karol Hausman. Dynamics-aware unsupervised discovery of skills. In *International Conference on Learning Representations*, 2020.

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pp. 5026–5033. IEEE, 2012.

Srinivas Venkattaramanujam, Eric Crawford, Thang Doan, and Doina Precup. Self-supervised learning of distance functions for goal-conditioned reinforcement learning. *arXiv preprint arXiv:1907.02998*, 2019.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pp. 9929–9939. PMLR, 2020.

Haoqi Yuan and Zongqing Lu. Robust task representations for offline meta-reinforcement learning via contrastive learning. In *International Conference on Machine Learning*, pp. 25747–25759. PMLR, 2022.

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.

A ALGORITHM

In this section, we provide the algorithmic procedure of CURE.

As illustrated in Algorithm 1, the algorithm consists of a contrastive pretraining phase and a main skill discovery phase. In the pretraining phase, we optimize the state encoder ϕ on an unlabeled dataset $\mathcal{D}_{\text{exp}}$ using the InfoNCE loss in Eq. 6 (line 4). We additionally enforce the Lipschitz constraint $\|\phi(s') - \phi(s)\|_2 \leq 1$ via a dual Lagrange multiplier (line 5). This pretraining stage injects a robust geometric prior into the representation space before policy learning starts, as analyzed in Section 4.2 and validated in Section 5. During the main skill discovery phase, we initialize the encoder with the pretrained parameters and then jointly optimize the encoder, policy, and Lagrange multiplier following the standard METRA procedure (line 11∼14). We protect the pretrained structure by adaptively controlling the encoder learning rate (line 13), as detailed in Section 4.3. Specifically, when the estimated state coverage is low, we use a reduced learning rate to prevent aggressive updates from overwriting the pretrained representation. Once the agent reliably traverses the environment, we restore the normal learning rate for fine-tuning.

Algorithm 1 CURE

Require: Exploratory dataset $\mathcal{D}_{\text{exp}}$, pretraining epochs T^{pre}, main training epochs T^{main}, number of negatives K, slack δ

1: Initialize encoder ϕ, Lagrange multiplier $\lambda_{\text{pre}}, \lambda$, replay buffer $\mathcal{B}$
2: **for** $e = 1$ **to** T^{pre} **do**
3: **for** each minibatch from $\mathcal{D}_{\text{exp}}$ **do**
4: Sample anchors (s, s'), positives from same trajectory, K negatives from other trajectories
5: Compute InfoNCE loss $\mathcal{L}_{\text{NCE}}$ via Eq. 6 and constraint $C = \min\big(\delta,\ 1 - \mathbb{E}[\|\Delta\phi\|_2^2]\big)$
6: Compute $\mathcal{L}_\phi \leftarrow \mathcal{L}_{\text{NCE}} - \lambda_{\text{pre}} \cdot C, \quad \mathcal{L}_{\lambda_{\text{pre}}} \leftarrow \log(\lambda_{\text{pre}}) \cdot C$
7: Update ϕ to minimize $\mathcal{L}_\phi$, update λ_{pre} to minimize $\mathcal{L}_\lambda$
8: **end for**
9: **end for**
10: **for** $e = 1$ **to** T^{main} **do**
11: Sample skill $z \sim p(z)$, execute $\pi(a|s, z)$ in the environment, store transitions in $\mathcal{B}$
12: Sample transitions from $\mathcal{B}$
13: Update ϕ using Eq. 2 with the learning rate calculated as Section 4.3
14: Update λ using Eq. 3, update π to maximize intrinsic reward in Eq. 4
15: **end for**

B Extended Related Work

Unsupervised skill discovery. Unsupervised skill discovery aims to learn a set of distinguishable behaviors that cover the state space without task-specific rewards. Early methods (Eysenbach et al., 2019; Sharma et al., 2020; Liu & Abbeel, 2021; Laskin et al., 2022) maximize the mutual information between latent skills and states to encourage behavioral diversity. However, these approaches often yield static behaviors and lack broad exploration (Park et al., 2022; 2024). Distance-maximizing skill discovery methods (Park et al., 2023) address this limitation by aligning distances in the representation space with distances in state space. For example, (Park et al., 2022) uses Euclidean distance as the distance metric to encourage geometrically longer travel. (Park et al., 2023) maximizes the negative log-likelihood of transition probabilities to prioritize rarely visited states. (Park et al., 2024) utilizes temporal distance metrics to encourage temporally distant state transitions. Yet, as analyzed in Section 3, these methods could struggle in environments with irreversible failure modes.

Prior-guided skill discovery. To further steer unsupervised skill discovery toward desirable behaviors, recent work (Kim et al., 2024; 2023; Hussonnois et al., 2023; 2025) incorporates prior knowledge into the learning process. For example, (Kim et al., 2024) trains a classifier from video data to distinguish desirable and undesirable transitions and scales the intrinsic reward accordingly. (Kim et al., 2023) applies analytical constraints via Lagrangian optimization to ensure safety. (Hussonnois et al., 2023; 2025) use pairwise human preferences to align skills with human values. Our work shares this fundamental motivation of injecting prior knowledge to expand the capabilities of skill discovery. Specifically, we introduce a geometric prior through contrastive pretraining to structurally resolve the suboptimal entrapment issue.

Contrastive learning for reinforcement learning. Contrastive learning extracts meaningful features by aligning positive samples and separating negative samples (Oord et al., 2018; Chen & He, 2021). This paradigm improves sample efficiency in image-based reinforcement learning (Laskin et al., 2020; Yuan & Lu, 2022) and facilitates exploration in goal-conditioned settings (Eysenbach et al., 2022; Liu et al., 2025). Within unsupervised skill discovery, (Laskin et al., 2022) optimizes contrastive objectives on online interaction data to maximize mutual information. Furthermore, theoretical analysis confirms that distance-maximizing objectives closely align with contrastive lower bounds (Zheng et al., 2025), which intuitively supports that contrastive objectives can effectively extract structural representations for skill discovery. Our work reaches a similar conclusion from a different perspective. We employ contrastive learning as a pretraining stage to establish a geometrically meaningful representation before skill discovery begins, which provably mitigates the suboptimal entrapment issue.

C Proofs and Additional Theoretical Analysis

C.1 Notations

For clarity, we summarize the key notations used throughout our theoretical analysis in Table 1.

Table 1: Organization of the notations used in the theoretical analysis.

Notation	Explanation
$\mathcal{S}, \mathcal{A}$	State space, action space
H	Episode horizon
$\mathcal{S}_{\text{succ}}, \mathcal{S}_{\text{abs}}$	Sets of success and absorbing states in the abstract MDP
s_h, x_h	A success state and its corresponding absorbing state at progression level h
a^*, a_{sub}	Optimal and suboptimal abstract actions
$\mathcal{L}_{\text{metra}}$	METRA's Lagrangian objective
π	Policy
ϕ	State representation (encoder)
z	Latent skill vector
$\mathcal{D}$	Initial exploration dataset
N	Number of exploration trajectories in the initial dataset
p_0	Upper bound of the optimal action probability
Q^ϕ, V^ϕ	Q-value and state-value functions under encoder ϕ
$\mathcal{M}_{\text{sub}}$	Suboptimal invariant set (absorbing set)
Σ_{exp}	Temporal transition covariance matrix of $\mathcal{D}_{\text{exp}}$
$U(\theta)$	Potential function defined as $-\mathcal{L}_{\text{metra}}(\theta)$
$\mathcal{B}$	Potential barrier height

C.2 Proof of Proposition 3.1

First, we formalize the notion of the absorbing set.

Definition C.1 (Absorbing set)**.** *A non-trivial proper subset $\mathcal{M} \subsetneq \mathcal{S}$ is an absorbing set if $P(s' \in \mathcal{M} \mid s \in \mathcal{M}, a) = 1$ holds for all $s \in \mathcal{M}$ and $a \in \mathcal{A}$. An absorbing state is a special case where the absorbing set is a singleton.*

We would like to note that an absorbing set may contain internal transitions. Our theoretical analysis of suboptimal entrapment focuses on the non-degenerate case where the trapped agent can still move and receive intrinsic rewards inside the absorbing set.

We then formalize the notion of temporal distance (Venkattaramanujam et al., 2019).

Definition C.2 (Temporal distance)**.** *For any two states $u, v \in \mathcal{S}$, the temporal distance $d_{temp}(u, v)$ is defined as the minimum number of environment steps required to transition from u to v. If v is unreachable from u, $d_{temp}(u, v) = \infty$.*

Then, we reformalize the environment as a Generalized Absorbing MDP, defined as follows.

Definition C.3 (Generalized Absorbing MDP)**.** *A Generalized Absorbing MDP is a finite-horizon chain MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, s_0, H)$:*

- *State space abstraction: $\mathcal{S} = \mathcal{S}_{succ} \cup \mathcal{S}_{abs}$, where $\mathcal{S}_{succ} = \{s_0, s_1, \ldots, s_H\}$ denotes the sequence of "success" states, and $\mathcal{S}_{abs} = \{x_0, x_1, \ldots, x_{H-1}\}$ denotes the corresponding "absorbing" states. Here, each s_h and x_h represents a semantic aggregation of underlying MDP states sharing the same macroscopic progression level h.*
- *Action space abstraction: At each $s_h \in \mathcal{S}_{succ}$, the action set is aggregated into two abstract actions: a^*, the "optimal" action that maintains progress, and a_{sub}, any "suboptimal" action that leads to failure.*
- *Transition dynamics: For all $0 \le h < H$, $P(s_{h+1} \mid s_h, a^*) = 1$, $P(x_h \mid s_h, a_{sub}) = 1$, $P(x_h \mid x_h, \cdot) = 1$ (absorbing).*

- *Temporal distance:* $d_{temp}(s_h, s_{h+1}) = 1$ *and* $d_{temp}(s_h, x_h) = 1$.

We now provide the proof of Prop. 3.1. We consider the beginning of the training process in the Generalized Absorbing MDP. An agent collects N exploration trajectories with a policy π_{exp}. This forms the initial dataset $\mathcal{D}$. At the beginning of training, before the agent has learned to progress, this policy is unlikely to make consistent forward progress. The analysis in this section holds under the following mild assumptions: (1) The exploration policy π_{exp} selects the optimal action with a probability strictly less than 1, i.e., $\exists p_0 \in (0, 1)$ such that $\mathbb{P}_{\pi_{\text{exp}}}(a = a^* \mid s_h) \leq p_0$. (2) For unvisited transitions, the encoder produces difference vectors with a bounded norm $\|\phi(s_{h+1}) - \phi(s_h)\|_2 \leq \epsilon_0 < 1$, and the value function for such states is bounded by $|V(s, z)| \leq \epsilon_V$, such that $\epsilon_0 + \gamma\epsilon_V < 1$.

Proposition C.1 (Prop. 3.1, restated)**.** *Consider the Generalized Absorbing MDP and the assumptions stated above. An agent collects N exploration trajectories with policy π_{exp} to form an initial dataset $\mathcal{D}$. For any integer L with $0 \leq L < H$, with probability at least $(1-p_0^{L+1})^N \geq 1 - Np_0^{L+1}$, the deterministic policy π_z subsequently optimized by METRA fails to reach state s_{L+1}.*

Proof. Since the probability of transitioning forward by action a^* is at most p_0, the probability of a single trajectory reaching s_{L+1} is bounded by p_0^{L+1}. Across N independent trajectories, the probability that no trajectory reaches s_{L+1} is bounded by Bernoulli's inequality:

$$\mathbb{P}(s_{L+1} \notin \mathcal{D}) \geq (1 - p_0^{L+1})^N \geq 1 - N \cdot p_0^{L+1}. \tag{8}$$

Thus, with high probability, the deepest non-absorbing transition in $\mathcal{D}$ stops at s_l where $l \leq L$. We analyze the boundary case where $\mathcal{D}$ contains visits to s_L but none to s_{L+1}. In this case, At state s_L, the dataset $\mathcal{D}$ contains transitions $(s_L, a_{\text{sub}}, x_L)$ but contains no transitions for (s_L, a^*, s_{L+1}).

METRA maximizes the objective $\max_\phi \mathbb{E}_{\mathcal{D}}[(\phi(s') - \phi(s))^\top z]$ subject to the Lipschitz constraint $\|\phi(s') - \phi(s)\|_2 \leq 1$. For the observed transition (s_L, x_L), the optimizer will align the difference vector with the skill vector z to maximize the inner product. Therefore, at the optimum ϕ^*:

$$\phi^*(x_L) - \phi^*(s_L) = z. \tag{9}$$

The resulting intrinsic reward for this transition is $r(s_L, x_L, z) = (\phi^*(x_L) - \phi^*(s_L))^\top z = z^\top z = 1$, since $\|z\|_2 = 1$.

Then, we evaluate the Q-values for the two abstract actions at s_L using the learned encoder ϕ^* and the Bellman equation $Q(s, a, z) = r(s, a, s') + \gamma V(s', z)$.

Specifically, for a_{sub}, the transition is (s_L, a_{sub}, x_L). Since x_L is absorbing, $V^\phi(x_L, z) = 0$. Thus, $Q^\phi(s_L, a_{sub}, z) = 1 + 0 = 1$.

For a^*, the transition is (s_L, a^*, s_{L+1}). Since (s_L, s_{L+1}) is not in $\mathcal{D}$, its intrinsic reward $r(s_L, s_{L+1}, z) = (\phi^*(s_{L+1}) - \phi^*(s_L))^\top z \leq \|\phi^*(s_{L+1}) - \phi^*(s_L)\|_2 \cdot \|z\|_2 \leq \epsilon_0$. Also, $V^\phi(s_{L+1}, z) \leq \epsilon_V$ as s_{L+1} is an unvisited state. Thus, we have $Q^\phi(s_L, a^*, z) \leq \epsilon_0 + \gamma\epsilon_V$.

Comparing the Q-values, we have:

$$Q^\phi(s_L, a_{sub}, z) - Q^\phi(s_L, a^*, z) \geq 1 - (\epsilon_0 + \gamma\epsilon_V) > 0. \tag{10}$$

This completes the proof. □

Discussion on the generality of the suboptimal entrapment issue. Although Prop. C.1 is stated for the METRA objective, the underlying mechanism is not specific to a particular choice of the distance constraint. More broadly, distance-maximizing skill discovery (DSD) methods jointly learn a state representation ϕ and a skill-conditioned policy π to maximize an objective of the form $\mathbb{E}[\sum_t (\phi(s_{t+1}) - \phi(s_t))^\top z]$ under the constraint $\|\phi(x) - \phi(y)\| \leq d(x, y)$ for all $x, y \in \mathcal{S}$. The coupling between ϕ and π creates a self-reinforcing loop: when early exploration encounters an absorbing set, the transitions into that set become overrepresented in the replay buffer. The encoder then adapts to align the absorbing transitions with the skill direction z to maximize the inner-product reward, thereby distorting the reward landscape in favor of failure behaviors. The policy, optimized

against this distorted reward, is further driven toward absorbing states, which in turn supplies the encoder with even more absorbing transitions. This feedback cycle can lock the joint optimization into a suboptimal stationary point, as analyzed in the following Prop. C.2. Therefore, this intuition and the empirical evidence (Section 5) suggest that similar issues are likely to arise in DSD methods whose distance metric is static or predefined by the environment, where representation learning and policy optimization are coupled through an intrinsic reward derived from a similar representation. Our analysis of METRA serves as a clean analytical case study of this phenomenon. The subsequent analysis indicates more properties of this issue.

C.3 Proof of Proposition 3.2

We first define the notion of a suboptimal invariant set. This denotes a set of states from which a particular policy cannot escape. We note that the absorbing set $\mathcal{M}$ defined in Section 2.3 is a suboptimal invariant set, where any deterministic policy can serve as the π_{sub} in the following assumption.

Definition C.4 (Suboptimal Invariant Set)**.** *A set of states $\mathcal{M}_{sub} \subset \mathcal{S}$ is a suboptimal invariant set for a skill z if there exists a stochastic policy $\pi_{sub}(\cdot|s,z)$ with support $S_{sub}(s) \triangleq \operatorname{supp} \pi_{sub}(\cdot|s,z)$ such that, for all $s \in \mathcal{M}_{sub}$ and for all actions within the support, the next state s' remains in $\mathcal{M}_{sub}$ with probability 1. Formally,*

$$\forall s \in \mathcal{M}_{sub},\ \forall a \in S_{sub}(s), \quad P(s' \in \mathcal{M}_{sub} \mid s, a) = 1. \tag{11}$$

We say $\mathcal{M}_{sub}$ is non-degenerate if for all $s \in \mathcal{M}_{sub}$, there exists $s' \in \mathcal{M}_{sub}, s' \neq s$ with positive probability under π_{sub}.

The METRA objective is optimized via a Lagrangian to handle the Lipschitz constraint on the encoder ϕ. [2]

$$\mathcal{L}_{\text{metra}}(\pi_\theta, \phi_\omega, \lambda) = \mathbb{E}_{\tau\sim\pi,z}\left[\sum(\phi(s_{t+1}) - \phi(s_t))^\top z\right] + \lambda\left(1 - \mathbb{E}_{\mathcal{D}}[\|\phi(s') - \phi(s)\|_2^2]\right). \tag{12}$$

We now provide the proof of Prop. 3.2, formulating the convergence of METRA to a suboptimal stationary point. We assume the existence of a suboptimal invariant set $\mathcal{M}_{\text{sub}} \subset \mathcal{S}$ and a corresponding policy π_{sub}. We consider a given skill vector z satisfying $\|z\|_2 = 1$. We assume the encoder ϕ_{sub} satisfies three properties under the state distribution induced by π_{sub}. To maximize the METRA objective, the encoder fully utilizes the Lipschitz budget for transitions within the invariant set, and the encoder difference aligns perfectly with the skill vector z. This means $\|\phi_{\text{sub}}(s') - \phi_{\text{sub}}(s)\|_2 = 1$ and $\phi_{\text{sub}}(s') - \phi_{\text{sub}}(s) = z$ for all $(s, s') \sim \pi_{\text{sub}}$. Also, the expected intrinsic reward outside the invariant set $\mathcal{M}_{\text{sub}}$ is strictly smaller by a margin $\delta > 0$. This means $\mathbb{E}_{s\notin\mathcal{M}_{\text{sub}},a\sim\pi}[(\phi_{\text{sub}}(s') - \phi_{\text{sub}}(s))^\top z] \leq 1 - \delta$.

Proposition C.2 (Prop. 3.2, restated)**.** *Assume a non-degenerate suboptimal invariant set $\mathcal{M}_{sub}$ and a trapped policy π_{sub} exist as described above. Under the aforementioned conditions, the tuple $(\pi_{sub}, \phi_{sub}, \lambda^* = 0.5)$ asymptotically satisfies the stationarity conditions of the METRA Lagrangian as the escaping probability mass vanishes. Specifically, it satisfies the KKT conditions: $\nabla_\theta \mathcal{L}_{metra} = \nabla_\omega \mathcal{L}_{metra} = \nabla_\lambda \mathcal{L}_{metra} = 0$.*

Proof. First, we analyze the stability with respect to λ. The gradient with respect to λ is:

$$\nabla_\lambda \mathcal{L}_{\text{metra}} = 1 - \mathbb{E}_{\pi_{\text{sub}}}\left[\|\phi_{\text{sub}}(s') - \phi_{\text{sub}}(s)\|_2^2\right]. \tag{13}$$

Under the assumptions, the squared norm is identically 1 for all transitions under π_{sub}. Therefore, $\nabla_\lambda \mathcal{L}_{\text{metra}} = 1 - 1 = 0$. This also implies that the constraint is active.

Then, we analyze the stability with respect to encoder parameters ω. Let $\Delta\phi = \phi_\omega(s') - \phi_\omega(s)$. The gradient with respect to ω is:

[2] In practice, METRA (Park et al., 2024) and similar methods (Park et al., 2022; Zheng et al., 2025) relax the per-transition constraint $\|\phi(s') - \phi(s)\|_2 \leq 1$ to an expectation over the replay buffer, $\mathbb{E}_{\mathcal{D}}[\|\phi(s') - \phi(s)\|_2^2] \leq 1$, which is enforced via dual gradient descent on the Lagrangian above.

$$\begin{aligned}\nabla_\omega \mathcal{L}_{\text{metra}} &= \mathbb{E}_{\pi_{\text{sub}},z}\left[\nabla_\omega(\Delta\phi^\top z)\right] - \lambda \cdot \mathbb{E}_{\pi_{\text{sub}},z}\left[\nabla_\omega(\|\Delta\phi\|_2^2)\right] \\ &= \mathbb{E}_{\pi_{\text{sub}},z}\left[(\nabla_\omega\Delta\phi)^\top z\right] - 2\lambda \cdot \mathbb{E}_{\pi_{\text{sub}},z}\left[(\nabla_\omega\Delta\phi)^\top\Delta\phi\right] \\ &= \mathbb{E}_{\pi_{\text{sub}},z}\left[(\nabla_\omega\Delta\phi)^\top(z - 2\lambda\Delta\phi)\right]\end{aligned} \tag{14}$$

Under the assumptions, $\Delta\phi_{sub} = z$. Substituting this yields $z - 2\lambda^* z = (1 - 2\lambda^*)z = 0$. Setting $\lambda^* = 0.5$ guarantees that the term in parentheses is zero, thus $\nabla_\omega \mathcal{L}_{\text{metra}} = 0$.

Finally, we analyze the stability with respect to policy parameters θ. By the Policy Gradient Theorem (Sutton et al., 1999), the gradient is given by the advantage function $A^{\phi_{sub}}(s, a, z)$:

$$\nabla_\theta \mathcal{L}_{\text{metra}} = \mathbb{E}_{s\sim d^{\pi_\theta}}\left[\sum_a \nabla_\theta \pi_\theta(a|s, z) \cdot A^{\phi_{sub}}(s, a, z)\right]. \tag{15}$$

For any $s \in \mathcal{M}_{\text{sub}}$, we partition the action space into inside-support actions $S_{\text{sub}}(s)$ and outside-support actions $S^c_{\text{sub}}(s)$. Let $Q^{\phi_{\text{sub}}}(s, a, z) = \mathbb{E}_{\pi_{\text{sub}}}\left[\sum_{t=0}^\infty \gamma^t r_t \mid s_0 = s, a_0 = a\right]$ denote the action-value function, where the intrinsic reward is $r_t = (\phi_{\text{sub}}(s_{t+1}) - \phi_{\text{sub}}(s_t))^\top z$. The state-value function is given by $V^{\phi_{\text{sub}}}(s, z) = \sum_a \pi_{\text{sub}}(a|s, z)Q^{\phi_{\text{sub}}}(s, a, z)$. Under the encoder ϕ_{sub}, we analyze the advantage function $A^{\phi_{\text{sub}}}(s, a, z) = Q^{\phi_{\text{sub}}}(s, a, z) - V^{\phi_{\text{sub}}}(s, z)$ for the two partitions:

Inside-support actions ($a \in S_{\text{sub}}(s)$): By Definition C.4, taking any $a \in S_{\text{sub}}(s)$ guarantees $s_t \in \mathcal{M}_{\text{sub}}$ for all $t \geq 0$. Based on our assumption, inside $\mathcal{M}_{\text{sub}}$, the encoder perfectly aligns with the skill vector, yielding persistent maximal rewards $r_t = z^\top z = 1$. Thus, the expected return is exactly the theoretical maximum:

$$Q^{\phi_{\text{sub}}}(s, a, z) = \sum_{t=0}^\infty \gamma^t \cdot 1 = \frac{1}{1-\gamma}, \quad \forall a \in S_{\text{sub}}(s). \tag{16}$$

Since π_{sub} assigns non-zero probabilities only to actions in $S_{\text{sub}}(s)$, the state-value function evaluates to:

$$V^{\phi_{\text{sub}}}(s, z) = \sum_{a\in S_{\text{sub}}(s)} \pi_{\text{sub}}(a|s, z)\left(\frac{1}{1-\gamma}\right) = \frac{1}{1-\gamma}. \tag{17}$$

Consequently, all actions in the support share the identical optimal value, yielding a zero advantage:

$$A^{\phi_{\text{sub}}}(s, a, z) = Q^{\phi_{\text{sub}}}(s, a, z) - V^{\phi_{\text{sub}}}(s, z) = 0. \tag{18}$$

Outside-support actions ($a \notin S_{\text{sub}}(s)$): Taking action $a \notin S_{\text{sub}}(s)$ risks escaping $\mathcal{M}_{\text{sub}}$. By our assumption, any transition escaping the invariant set degrades the expected immediate intrinsic reward to at most $1 - \delta$ ($\delta > 0$). Even assuming the most conservative upper bound where the agent collects the maximal reward $r = 1$ for all subsequent steps $t \geq 1$, the action-value is strictly bounded:

$$Q^{\phi_{\text{sub}}}(s, a, z) \leq (1 - \delta) + \sum_{t=1}^\infty \gamma^t \cdot 1 = (1 - \delta) + \frac{\gamma}{1-\gamma} = \frac{1}{1-\gamma} - \delta. \tag{19}$$

Therefore, the advantage function for these out-of-support actions is strictly negative:

$$A^{\phi_{\text{sub}}}(s, a, z) \leq \left(\frac{1}{1-\gamma} - \delta\right) - \frac{1}{1-\gamma} = -\delta < 0. \tag{20}$$

Decomposing the policy gradient yields:

$$\nabla_\theta \mathcal{L}_{\text{metra}} = \mathbb{E}_{s\sim d^{\pi_\theta}}\left[\sum_{a\notin S_{\text{sub}}(s)} \nabla_\theta \pi_\theta(a|s, z) \cdot A^{\phi_{\text{sub}}}(s, a, z)\right]. \tag{21}$$

Let $\epsilon(s) \triangleq \sum_{a\notin S_{\text{sub}}(s)} \pi_\theta(a|s, z)$ denote the total escaping probability mass. As $\epsilon(s) \to 0$, we have $\pi_\theta(a|s, z) \to 0$ for all $a \notin S_{\text{sub}}(s)$, which implies $\nabla_\theta \pi_\theta(a|s, z) \to 0$ under standard parameterized policy classes (e.g., Softmax). Consequently, $\|\nabla_\theta \mathcal{L}_{\text{metra}}\| \to 0$, establishing that π_{sub} is an asymptotic local stationary point.

For policy parameterizations that allow exact zero probability on out-of-support actions, the limit is an exact stationary point. For finite-parameter Softmax policies, zero probability is attained only asymptotically; hence, the policy-gradient stationarity condition should be understood in the limiting sense as $\epsilon(s) \to 0$.

Combining the above analysis completes the proof. □

Generalization of Prop. C.2 for the DSD framework. We now extend the analysis of Prop. C.2 to the Distance-maximizing Skill Discovery (DSD) framework. The DSD objective is as follows:

$$\max_{\pi,\phi} \mathbb{E}_{\tau\sim\pi,z}\left[\sum_t (\phi(s_{t+1}) - \phi(s_t))^\top z\right] \quad \text{s.t.} \quad \|\phi(x) - \phi(y)\|_2 \le d(x,y),\ x,y \in \mathcal{S} \tag{22}$$

Here, we restrict our attention to DSD methods with a static distance metric, where $d(\cdot,\cdot)$ is predefined by the environment and does not dynamically adapt to the policy's current competence, e.g., Euclidean distance in LSD (Park et al., 2022) or temporal distance in METRA (Park et al., 2024). While prior works enforce this constraint via different mechanisms, e.g., Lagrange multipliers in METRA (Park et al., 2024) and spectral normalization in LSD (Park et al., 2022), the first-order optimality conditions at convergence remain invariant. Consider a policy that has converged to a suboptimal invariant set $\mathcal{M}_{\text{sub}}$. Within the state distribution $\rho_\pi(s)$ induced by this policy, the representation optimization solves a constrained inner-product maximization, and the optimal encoder must align the transition difference $\Delta\phi_{s,s'} = \phi(s') - \phi(s)$ with the skill vector z and saturate the local constraint bound, i.e., $\|\Delta\phi_{s,s'}\| = d(s,s')$ for all visited transitions. At this configuration, the gradient of the objective with respect to ϕ becomes collinear with the gradient of the constraint function:

$$\nabla_\phi \mathbb{E}_{\rho_\pi}[(\Delta\phi)^\top z] = \kappa \cdot \nabla_\phi \mathbb{E}_{\rho_\pi}[\|\Delta\phi\|^2], \quad \kappa > 0. \tag{23}$$

Consequently, the effective update direction for ϕ on the state distribution vanishes, forming a stationary point for the representation module. Simultaneously, for the policy π, the intrinsic reward on transitions within $\mathcal{M}_{\text{sub}}$ reaches its theoretical maximum under the distance constraint. Also, any action attempting to leave $\mathcal{M}_{\text{sub}}$ or deviate from the current trajectory yields a strictly lower expected intrinsic reward. Thus, the policy advantage function satisfies $A^\phi(s,a,z) \le 0$ for all a, making the policy gradient $\nabla_\theta J = \mathbb{E}_{s\sim\rho_\pi}[\nabla_\theta \log \pi_\theta(a|s,z) A^\phi(s,a,z)]$ zero. The joint optimization dynamics therefore settle at a stationary point of the constrained DSD objective. This analysis implies that escaping requires overcoming a potential barrier in the parameter space, which we characterize in the following proposition.

C.4 Proof of Proposition 3.3

To characterize the difficulty of escaping the suboptimal stationary point, we model the parameter optimization landscape with a potential function $U(\theta) = -\mathcal{L}_{\text{metra}}(\theta)$. Here, $\mathcal{L}_{\text{metra}}$ is the Lagrangian objective of METRA. We analyze a suboptimal stationary point θ_{sub} located in a local neighborhood $\Omega_{\text{sub}} \subset \Theta$. This analysis assumes a locally μ-strongly convex potential within Ω_{sub} and a potential barrier $\mathcal{B}$ at its boundary. This means $U(\theta) \ge U(\theta_{\text{sub}}) + \frac{\mu}{2}\|\theta - \theta_{\text{sub}}\|_2^2$ for all $\theta \in \Omega_{\text{sub}}$. We define the barrier height $\mathcal{B}$ as the minimum potential difference at the boundary $\partial\Omega_{\text{sub}}$. This gives $\mathcal{B} = \inf_{\theta\in\partial\Omega_{\text{sub}}} U(\theta) - U(\theta_{\text{sub}})$. Consistent with our reward structure, the barrier height is bounded below by the cumulative reward gap between the trapped and an escaping policy, i.e., $\mathcal{B} \ge \alpha H \delta$. Here, H is the episode horizon. The constant $\alpha \in (0,1)$ is the expected proportion of steps in an episode spent within the suboptimal invariant set $\mathcal{M}_{\text{sub}}$, and αH steps are spent inside $\mathcal{M}_{\text{sub}}$ per episode. The constant $\delta \in (0,1)$ is the per-step intrinsic reward drop defined in Prop. C.2. Finally, we model the parameter update as stochastic gradient descent on U. We have $\theta_{t+1} = \theta_t - \eta\nabla_\theta U(\theta_t) + \eta\xi_t$. Here, $\xi_t \sim \mathcal{N}(0, \sigma^2 I)$ is Gaussian noise with variance $\sigma^2 > 0$, and η is the learning rate.

Proposition C.3 (Prop. 3.3, restated)**.** *Under the landscape and optimization model described above, we consider the small noise regime where $\eta\sigma^2 \ll \mathcal{B}$. The expected number of steps $\mathbb{E}[T_{escape}]$ for the parameter θ_t to escape from Ω_{sub} satisfies the following lower bound:*

$$\mathbb{E}[T_{escape}] \ge C \cdot \exp\left(\frac{2\alpha H\delta}{\eta\sigma^2}\right), \tag{24}$$

where $C > 0$ is a constant independent of the noise parameters.

Proof. We consider the discrete parameter update as follows:

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta U(\theta_t) + \eta \xi_t. \tag{25}$$

For a sufficiently small learning rate η, the discrete Markov chain $\{\theta_t\}$ can be approximated by a continuous-time diffusion process. We map the discrete steps t to continuous time τ with a time rescaling $d\tau = \eta$. The update corresponds to the Euler-Maruyama discretization of the Overdamped Langevin Equation (OLE) (Roberts & Tweedie, 1996):

$$d\theta_\tau = -\nabla_\theta U(\theta_\tau) d\tau + \sqrt{2D_{\text{eff}}} dW_\tau, \tag{26}$$

where W_τ is a standard Wiener process (Roberts & Tweedie, 1996) in the parameter space.

The effective diffusion coefficient D_{eff} is determined by matching the variance of the discrete noise over one step, $\text{Var}(\eta \xi_t) = \eta^2 \sigma^2$, with the variance of the continuous noise over a time interval η, which is $2D_{\text{eff}}\eta$. Equating these gives:

$$D_{\text{eff}} = \frac{\eta \sigma^2}{2}. \tag{27}$$

Then, the problem of escaping from the local minimum of $U(\theta)$ at θ_{sub} over the barrier $\mathcal{B}$ can be analyzed using large deviations theory for diffusions (Freidlin-Wentzell theory) (Freidlin & Wentzell, 1998). According to Kramers' escape rate formula (Kramers, 1940), the expected first passage time $\mathbb{E}[T_{\text{escape}}]$ over a barrier of height $\mathcal{B}$ scales exponentially in the limit of small noise ($D_{\text{eff}} \ll \mathcal{B}$):

$$\mathbb{E}[T_{\text{escape}}] \asymp \exp\left(\frac{\mathcal{B}}{D_{\text{eff}}}\right)^{3} \tag{28}$$

Substituting the expression for the effective diffusion coefficient D_{eff}:

$$\mathbb{E}[T_{\text{escape}}] \asymp \exp\left(\frac{2\mathcal{B}}{\eta \sigma^2}\right). \tag{29}$$

Therefore, there exists a constant $C_1 > 0$ such that for sufficiently small noise variance σ^2:

$$\mathbb{E}[T_{\text{escape}}] \geq C_1 \cdot \exp\left(\frac{2\mathcal{B}}{\eta \sigma^2}\right). \tag{30}$$

Finally, we establish the lower bound for the barrier height $\mathcal{B}$ in the context of the METRA objective. From the definition of the suboptimal invariant set and the assumptions, a policy that stays within $\mathcal{M}_{\text{sub}}$ receives a per-step reward of ≈ 1. In contrast, a policy at the boundary of the basin $\partial\Omega_{\text{sub}}$ attempts to leave $\mathcal{M}_{\text{sub}}$ and consequently suffers a drop in expected reward $r \leq 1 - \delta$.

That is, αH is the number of steps the agent spends inside $\mathcal{M}_{\text{sub}}$. Thus, the difference in cumulative reward between the stationary policy and a boundary policy is at least $\alpha H(1 - (1 - \delta)) = \alpha H \delta$. Since the objective function $\mathcal{L}_{\text{metra}}$ is proportional to this cumulative reward, the potential barrier $\mathcal{B} = U(\theta_{\text{boundary}}) - U(\theta_{\text{sub}})$ satisfies:

$$\mathcal{B} = \mathcal{L}_{\text{metra}}(\theta_{\text{sub}}) - \mathcal{L}_{\text{metra}}(\theta_{\text{boundary}}) \geq \alpha \sum_{t=1}^{H} (r_{\text{sub}} - r_{\text{boundary}}) \geq \alpha H \delta. \tag{31}$$

Substituting the bound $\mathcal{B} \geq \alpha H \delta$ into the escape time formula yields:

$$\mathbb{E}[T_{\text{escape}}] \geq C \cdot \exp\left(\frac{2\alpha H \delta}{\eta \sigma^2}\right) \tag{32}$$

where $C > 0$ is a constant. This completes the proof. □

[3] The notation $\asymp$ denotes logarithmic equivalence. Specifically, $\lim_{D_{\text{eff}} \to 0} D_{\text{eff}} \ln \mathbb{E}[T_{\text{escape}}] = \mathcal{B}$.

Generalization of Prop. C.3 for the DSD framework. We now extend the analysis of Prop. C.3 to the Distance-maximizing Skill Discovery (DSD) framework, as formalized in Eq. 22. When the policy collapses into a suboptimal invariant set $\mathcal{M}_{\text{sub}}$ and converges, the encoder fully utilizes the constraint budget on those trapped transitions, yielding a near-maximal per-step reward. On the other hand, any attempt to deviate outside $\mathcal{M}_{\text{sub}}$ cannot fully exploit the constraint in the direction of z, and therefore suffers a lower reward. Under a static distance metric, such as those in (Park et al., 2024; 2022), no mechanism reduces the reward of the overrepresented failure transitions, so the optimization landscape still exhibits a potential barrier $\mathcal{B}$ that separates the collapsed behavior from other potentially higher-quality policies. The analysis in Prop. C.3 relies only on the existence of such a barrier and a reward gap. Thus, similar analysis results hold for these DSD methods.

C.5 Proof of Proposition 4.1

This subsection provides a formal connection between our proposed pretraining on given data and principal component analysis (PCA) (Hotelling, 1933) of the temporal transition covariance matrix. This analysis is conducted under a set of simplifying assumptions that allow for a tractable linear characterization, formalized as follows.

Definition C.5 (Temporally consistent embedding and linear encoder)**.** *Following the formulation of METRA (Appendix C, Definition C.1), we assume the existence of a temporally consistent embedding $\psi : \mathcal{S} \to \mathbb{R}^m$ such that $d_{temp}(u, v) = \|\psi(u) - \psi(v)\|_2$ for all state pairs u, v. Furthermore, the learned encoder $\phi : \mathcal{S} \to \mathbb{R}^d$ is assumed to be a linear projection of this embedding, i.e., $\phi(s) = W^\top \psi(s)$, where $W \in \mathbb{R}^{m\times d}$ is a projection matrix satisfying the Lipschitz constraint $\|W\|_{op} \leq 1$.*

Intuitively, this implies that the temporal distance metric can be embedded into a (potentially high-dimensional) Euclidean space, and the encoder can be expressed as a linear projection of this embedding. We would like to note that this linear formulation is compatible with a standard nonlinear MLP encoder by viewing $\psi(s)$ as the feature representation produced by its penultimate layer. The final layer then applies the linear projection $W^\top$, yielding $\phi(s) = W^\top \psi(s)$. Thus, only the final projection is assumed to be linear, while the overall encoder can remain nonlinear.

We next define the temporal transition covariance matrix.

Definition C.6 (Temporal transition covariance matrix and linear encoder)**.** *For a distribution $p(\tau)$ over trajectories, the temporal transition covariance matrix $\Sigma_p \in \mathbb{R}^{m\times m}$ is defined as:*

$$\Sigma_p = \mathbb{E}_{(s,s')\sim p(\tau)} \left[(\psi(s') - \psi(s))(\psi(s') - \psi(s))^\top \right], \tag{33}$$

where $\psi : \mathcal{S} \to \mathbb{R}^m$ is the underlying temporal embedding. This implies that the temporal distance metric can be embedded into a Euclidean space.

We denote the covariance matrix of the given dataset $\mathcal{D}_{\text{exp}}$ as Σ_{exp}. We assume the global expectation of temporal transitions in the given dataset $\mathcal{D}_{\text{exp}}$ is approximately zero, meaning $\mathbb{E}[\Delta\psi] \approx 0$. We additionally assume that transition displacements sampled as positive pairs from the same trajectory are directionally coherent, such that

$$\mathbb{E}[\Delta\psi(\Delta\psi^+)^\top] = \rho\, \Sigma_{\text{exp}}, \tag{34}$$

for some constant $\rho > 0$. This assumption formalizes the temporal directional coherence targeted by our positive-pair construction.

We provide the proof of Prop. 4.1 as follows.

Proposition C.4 (Prop. 4.1, restated)**.** *Under the linear encoder assumptions stated above, we optimize the InfoNCE loss over $\mathcal{D}_{exp}$. In the limit of a sufficiently large negative sample size ($K \to \infty$) and a sufficiently high temperature, minimizing the InfoNCE loss $\mathcal{L}_{NCE}$ is asymptotically equivalent to extracting the top-d principal components of Σ_{exp}.*

Proof. According to (Wang & Isola, 2020), as $K \to \infty$, the InfoNCE loss converges to:

$$\mathcal{L}_{\text{NCE}}(W) = -\left(\mathbb{E}\left[\frac{\Delta\phi^\top \Delta\phi^+}{\tau} \right] - \mathbb{E}_{\Delta\phi}\left[\log \mathbb{E}_{\Delta\phi^-}\left[\exp\left(\frac{\Delta\phi^\top \Delta\phi^-}{\tau} \right) \right] \right] \right), \tag{35}$$

where $\Delta\phi$ and $\Delta\phi^+$ are positive pairs (transitions from the same trajectory) and $\Delta\phi^-$ is a negative sample (a transition from a different trajectory).

Then, we have:

$$\begin{aligned}\mathbb{E}[\Delta\phi^\top\Delta\phi^+] &\approx \mathbb{E}[\Delta\phi^\top\Delta\phi] \\ &= \mathbb{E}[\text{tr}(W^\top\Delta\psi\Delta\psi^\top W)] \\ &= \text{tr}(W^\top\mathbb{E}[\Delta\psi\Delta\psi^\top]W) = \rho\text{tr}(W^\top\Sigma_{\text{exp}}W).\end{aligned} \tag{36}$$

Let $X = \frac{1}{\tau}\Delta\phi^\top\Delta\phi^-$. Since $\mathbb{E}[\Delta\phi^-] = 0$, we have $\mathbb{E}[X] = 0$. Using the second-order Taylor expansion of the cumulant generating function, $\log\mathbb{E}[e^X] \approx \mathbb{E}[X] + \frac{1}{2}\text{Var}(X)$, we have:

$$\begin{aligned}\mathbb{E}_{\Delta\phi}\left[\log\mathbb{E}_{\Delta\phi^-}[e^X]\right] &\approx \frac{1}{2\tau^2}\mathbb{E}_{\Delta\phi,\Delta\phi^-}\left[\text{tr}(\Delta\phi\Delta\phi^\top\Delta\phi^-(\Delta\phi^-)^\top)\right] \\ &= \frac{1}{2\tau^2}\text{tr}\left((W^\top\Sigma_{exp}W)^2\right).\end{aligned}$$

The approximated objective becomes:

$$\max_{\|W\|_{op}\le 1}\frac{\rho}{\tau}\text{tr}(W^\top\Sigma_{exp}W) - \frac{1}{2\tau^2}\text{tr}\left((W^\top\Sigma_{exp}W)^2\right). \tag{37}$$

Let λ_i be the eigenvalues of $W^\top\Sigma_{exp}W$. The objective aims to maximize $\sum_i(\frac{1}{\tau}\lambda_i - \frac{1}{2\tau^2}\lambda_i^2)$. For high temperatures $\tau \gg \lambda_i$, the linear term dominates the quadratic term. Thus, the optimization simplifies to $\max\text{tr}(W^\top\Sigma_{exp}W)$ subject to $\|W\|_{op} \le 1$. By the Ky Fan's Maximum Principle (Bhatia, 2013), the optimal matrix W_{pt} has columns that are the top-d eigenvectors of the covariance matrix Σ_{exp}. That is, InfoNCE pretraining on given data is asymptotically equivalent to performing PCA on Σ_{exp}, thereby completing the proof. □

C.6 Proof of Corollary 4.1

We apply the pretraining results to the Generalized Absorbing MDP setup from Prop. C.1. We consider the same setup where an initial exploration dataset $\mathcal{D}_{\text{init}}$ contains successful trajectories that pass the frontier s_L, and assume the given dataset $\mathcal{D}_{\text{exp}}$ contains successful trajectories traversing the chain to depth at least $L+1$. We further assume that the corresponding progressing displacement is sufficiently aligned with the first principal component. Specifically, orienting v_1 toward the progressing direction,

$$v_1^\top\Delta\psi_{\text{succ}} \ge 1 - \delta_{\text{pt}}.$$

for some small $\delta_{\text{pt}} \ge 0$.

Corollary C.1 (Cor. 4.1, restated)**.** *Consider the setup of Prop. C.1. Suppose the given dataset $\mathcal{D}_{exp}$ contains trajectories that successfully traverse the chain to depth at least $L+1$. Let ϕ_{pt} be the encoder obtained via InfoNCE pretraining on $\mathcal{D}_{exp}$, with $W_{pt} = [v_1, \ldots, v_d]$, where v_1 is the first principal component of Σ_{exp}, and let $z = e_1 \in \mathbb{R}^d$ be the corresponding skill vector in the representation space. Then, during the subsequent METRA training phase initialized with ϕ_{pt}, the Q-value gap at the state s_L is bounded. Specifically, there exists a small constant $\delta' \ll 1$ such that:*

$$Q^{\phi_{pt}}(s_L, a^*, z) \ge Q^{\phi_{pt}}(s_L, a_{sub}, z) - \delta'. \tag{38}$$

Proof. By Prop. C.4, $\phi_{\text{pt}}(s) = W_{\text{pt}}^\top\psi(s)$, where W_{pt} spans the top-d principal components of Σ_{exp}. By the principal-direction alignment assumption and $z = e_1$,

$$r_{\text{pt}}(s_L, s_{L+1}, z) = (\phi_{\text{pt}}(s_{L+1}) - \phi_{\text{pt}}(s_L))^\top z \ge 1 - \delta_{\text{pt}}. \tag{39}$$

Now, we evaluate the Q-values at the frontier state s_L using the pretrained encoder ϕ_{pt}. For the suboptimal action a_{sub}, the Lipschitz constraint still implies $r(s_L, x_L, z) \le 1$. Thus,

$$Q^{\phi_{\text{pt}}}(s_L, a_{\text{sub}}, z) \leq 1. \tag{40}$$

For the optimal action a^*, we have

$$Q^{\phi_{\text{pt}}}(s_L, a^*, z) \geq (1 - \delta_{\text{pt}}) + \gamma V(s_{L+1}, z) \geq (1 - \delta_{\text{pt}}) - \gamma\epsilon_V. \tag{41}$$

Let $\delta' = \delta_{\text{pt}} + \gamma\epsilon_V$ completing the proof. □

Interpretation of Cor. C.1. We would like to emphasize that Cor. C.1 does not claim that pretraining necessarily makes the progressing action a^* strictly preferable to the absorbing action a_{sub}. Its role is to show that pretraining removes the large Q-value bias that causes the entrapment identified in Prop. C.1. Specifically, Prop. C.1 establishes that, without pretraining,

$$Q^{\phi}(s_L, a_{\text{sub}}, z) - Q^{\phi}(s_L, a^*, z) \geq 1 - (\epsilon_0 + \gamma\epsilon_V) > 0, \tag{42}$$

so the policy is strongly biased toward the absorbing transition. In contrast, Cor. C.1 gives

$$Q^{\phi_{\text{pt}}}(s_L, a_{\text{sub}}, z) - Q^{\phi_{\text{pt}}}(s_L, a^*, z) \leq \delta', \tag{43}$$

where δ' is small. Thus, CURE reduces an $O(1)$ suboptimal Q-value gap to an $O(\delta')$ gap, substantially correcting the reward distortion responsible for suboptimal entrapment. The corollary is therefore intended to establish the removal of the strong preference for the absorbing action. This theoretical guarantee is also consistent with our empirical results, where the pretrained initialization substantially increases the probability of sustained forward exploration.

Table 2: Performance on zero-shot tasks.

Method	Mean ± Std	Max
CURE	3.44 ± 3.00	6.47
METRA	-0.74 ± 4.55	5.96
DoDont	1.38 ± 4.59	7.34
LSD	0.23 ± 4.90	7.31
DIAYN	-4.79 ± 0.93	3.43

Table 3: Performance on hierarchical control.

Method	Mean ± Std	Max
CURE	11.36 ± 8.11	32.62
METRA	4.83 ± 4.88	19.88
DoDont	6.20 ± 4.94	16.33
LSD	3.68 ± 3.56	14.29
DIAYN	0.23 ± 0.37	1.60

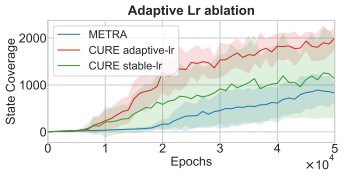


Figure 7: Ablation on the adaptive learning rate mechanism.

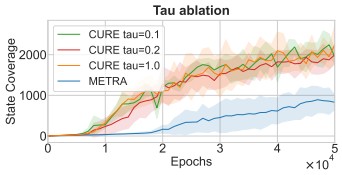


Figure 8: Ablation on the values of InfoNCE temperature.

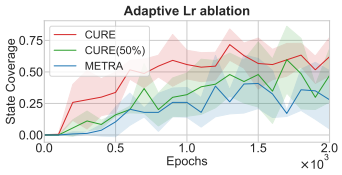


Figure 9: Trajectory mixture ablation in HalfSphere.

D Additional Experimental Results

D.1 Downstream Task Performance

We evaluate the utility of discovered skills of CURE and baselines via zero-shot and hierarchical control on downstream tasks. We define a motion task with a large falling penalty and a goal-reaching task in Humanoid, as detailed in Appendix E.3. As shown in Tables 2 and 3, CURE achieves better zero-shot performance on the motion task, and using its skills as low-level primitives yields higher cumulative rewards on the goal-reaching task than baseline skills. These results indicate that the improved exploration and robustness of CURE produce more capable and transferable skills for downstream control.

D.2 Additional Ablation Studies

Ablation on the failure-mixed dataset in the Halfsphere environment. We trained our model on a mixed dataset collected in the Halfsphere environment, where failed trajectories account for 50% of the total data. As shown in Fig. 9, although the inclusion of mixed trajectories leads to a performance drop, CURE still consistently outperforms the baseline.

Ablation on the adaptive learning rate mechanism. We examine the necessity of the exploration-triggered learning rate mechanism in Section 4.3 by training the encoder with a constant learning rate instead. As Fig. 7 shows, without this mechanism, training becomes unstable and final state coverage declines significantly. This indicates that a slow-to-fast schedule is critical for protecting the pretrained prior during the early stages of skill learning.

Sensitivity to InfoNCE's temperature. We evaluate the sensitivity of CURE to the InfoNCE temperature $\tau \in \{0.1, 0.2, 1.0\}$. Fig. 8 shows that the final state coverage is stable across these values, showing that CURE does not require precise temperature tuning to capture transition dynamics.

D.3 METRA's Behaviour at 200k Iterations

To empirically verify that prolonged training alone cannot help the policy escape this local optimum, we extend METRA training to 200k steps. As shown in Figure 10, although the policy learns diverse forward-locomotion skills, several latent skills remain trapped in the same failure behavior, which collapse near the origin shortly after initialization.

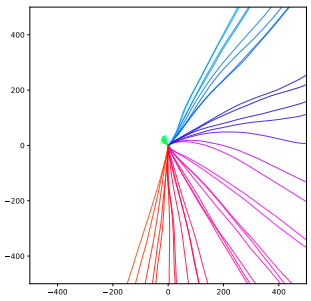


Figure 10: Learned skills by METRA after 200k training iterations on the x-y plane.

D.4 Visualizations of Learned Skills

To qualitatively assess exploration, we visualize the trajectories of learned skills on the x-y plane in Fig. 11. CURE achieves broad and uniform state coverage, effectively avoiding premature convergence to absorbing states.

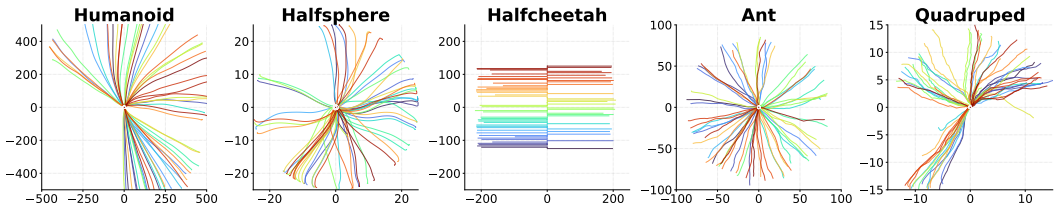
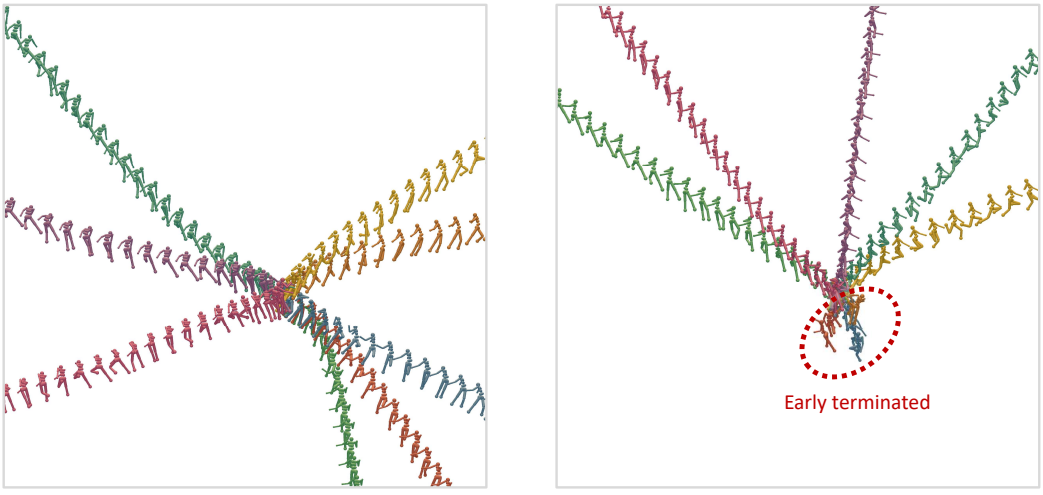


Figure 11: Visualization of the trajectories of learned skills on the x-y plane.

We further visualize the specific locomotion behaviors of individual Humanoid skills in Fig. 12. CURE successfully learns stable walking skills that move in diverse directions. In comparison, multiple skills trained by METRA collapse prematurely and fail to recover from falls, which restricts exploration and degrades skill diversity. These visualizations confirm that our approach mitigates the suboptimal entrapment issue and learns a more diverse set of behaviors.



(a) CURE's behavior visualization

(b) METRA's behavior visualization

Figure 12: Behavior visualization of CURE and METRA on Humanoid task, using uniformly sampled 8 skill vectors.

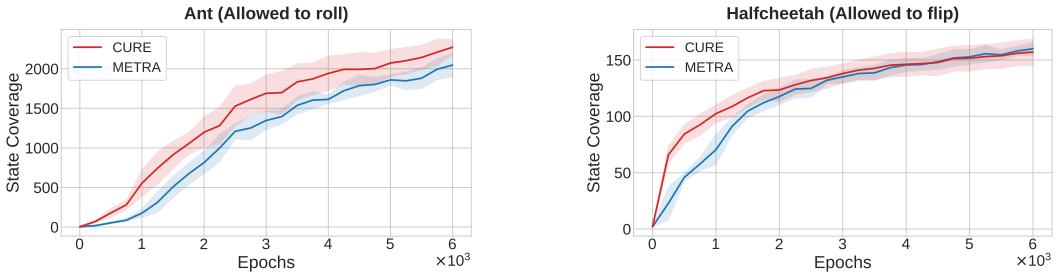


Figure 13: Training curves of Ant (allowed to roll) and Halfcheetah (allowed to flip).

D.5 Effectiveness on Environments Without Irreversible Failure Modes

We further evaluate CURE in environments without irreversible failure modes. Specifically, we disable early termination in the Ant and HalfCheetah domains to simulate standard learning settings. Fig. 13 presents the training curves for state coverage across these environments. CURE achieves comparable or superior performance relative to METRA. This result demonstrates the broad applicability of our pretraining framework beyond failure-prone scenarios.

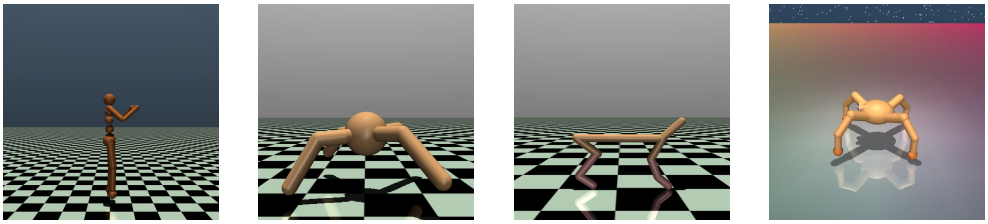
We attribute this acceleration to the geometric prior injected by contrastive pretraining. The pretrained encoder captures the principal components of the transition dynamics before policy learning begins. This initialization equips the skill discovery process with a representation space that already reflects meaningful state displacements. Consequently, the policy can produce directionally consistent behaviors from the early stages of training, rather than wasting samples on randomly drifting motions. The faster alignment between the skill latent space and the environment's underlying dynamics leads to more efficient maximization of the intrinsic distance reward. These results confirm that CURE offers a general and lightweight mechanism for accelerating distance-maximizing skill discovery, even in the absence of irreversible failure modes.

E EXPERIMENTAL DETAILS

In this section, we provide the complete details of our experimental setup, including the source of exploratory data, environment descriptions, evaluation metrics, downstream tasks, and hyperparameters used for our method and the baselines.

E.1 ENVIRONMENTS

We evaluate our method across a range of continuous control tasks. For state-based environments, we utilize MuJoCo-based *Humanoid*, *HalfCheetah*, and *Ant* (Todorov et al., 2012; Brockman et al., 2016). For the pixel-based environment, we evaluate *Quadruped* from the DeepMind Control Suite (DMC) (Tassa et al., 2018). These environments are visualized in Fig. 14.



(a) Humanoid (b) Ant (c) HalfCheetah (d) Quadruped

Figure 14: Visualization of evaluated environments. Illustration of HalfSphere is provided in Fig. 4.

To ensure the learning of stable and robust skills, we incorporate early termination criteria as follows:

- **Humanoid**: The episode terminates if the center of mass falls below 1.0 meters.
- **HalfCheetah**: The agent is terminated if the angle between its torso and the ground exceeds $45°$.
- **Ant & Quadruped**: Episodes are terminated if the agent flips over.

Additionally, we introduce a custom-designed **HalfSphere** environment implemented via the OpenAI Gym API (Brockman et al., 2016), as illustrated in Fig. 4. In this task, a mass point moves on the surface of a hemisphere with a radius of 25m under a gravitational acceleration of $g = 9.8$ m/s^2. The agent is controlled by an external force with a maximum magnitude of 10, applied tangentially to the spherical surface. Additionally, a constant centripetal force of 0.1 N is applied to the mass point, directed toward the center of the hemisphere. This inward radial force ensures that the equatorial boundary is kinematically accessible, as the mass point would otherwise be physically unable to reach the lower extremities of the surface. An episode is early terminated if the agent moves beyond the boundary of the hemisphere.

E.2 EVALUATION METRICS

Following the protocol established in METRA (Park et al., 2024), we characterize the effectiveness of our algorithm through three primary metrics:

- **State Coverage**: This metric is used to evaluate the diversity and exploration capacity of the learned skill policy $\pi(a|s, z)$. for Humanoid, Ant, Quadruped, we discretize the (x, y) coordinate space into a grid of 1×1 unit bins. For HalfCheetah, we discretize the x coordinate space into one-dimensional unit bins. For HalfSphere, we discretize the (x, y, z) coordinate space into a grid of $1 \times 1 \times 1$ unit bins. Then, this metric is defined as the total number of unique bins visited across evaluation rollouts of 48 uniformly sampled skills.
- **Survival Rate**: We define the survival rate to quantify the stability of the learned skills. An agent is considered to survive if it reaches for 100% of the maximum episode length in *Ant*, *Quadruped*, and *HalfCheetah*, or at least 75% in the *Humanoid* environment.

E.3 Downstream Tasks

We assess the transferability of the learned skills within the *Humanoid* environment through two distinct tasks:

1. *Zero-shot Motion Evaluation*: We evaluate the intrinsic quality of the skills by executing randomly sampled z from the skill space, measuring the resulting accumulated motion reward. For a given transition, the reward R_t is formulated to capture both the directional velocity and the stability of the agent:

$$R_t = \max(|v_x|, |v_y|) + R_{\text{survive}} \tag{44}$$

   where v_x and v_y denote the horizontal and lateral velocities calculated via finite differences of the agent's center of mass. The survival term R_{survive} yields a penalty of -5.0 if the agent's height z falls outside the viable range $[1.0, 2.0]$ or if a termination condition is met. Otherwise, R_{survive} provides a constant bonus of $+1.0$. This metric effectively quantifies the agent's ability to maintain balanced, high-speed locomotion across the discovered skill space $\mathcal{Z}$.

2. *Hierarchical Goal-Reaching*: We deploy a high-level controller to select skills from a discrete latent space. These low-level skill policies are then composed to complete a navigation task, requiring the agent to reach a target location $\mathbf{g} = (x_g, y_g)$, where the coordinates are randomly sampled from a uniform distribution $\mathcal{U}(-50, 50)$ at the beginning of each episode.

   The hierarchical architecture operates at two temporal scales:

   - **High-level Policy**: A meta-controller $\pi^h(z|s)$ selects a discrete skill z from the pretrained latent space. To ensure temporal abstraction, the high-level policy is invoked every $k = 150$ environment steps.
   - **Low-level Policy**: The frozen pretrained skill policy $\pi(a|s, z)$ executes the selected skill for the duration of the high-level step, mapping the latent code and current state to primitive actions.

   The episode terminates either after a maximum horizon of $1,000$ environment steps or when the humanoid's center of mass falls below 1 meter, indicating a failure state. Importantly, entering the target region does not terminate the episode. The target region is defined as a circular area with a radius of 5 meters centered at $\mathbf{g}$. At each environment step, the agent receives a sparse reward of 1.0 if its position lies within the target region and 0 otherwise, allowing the agent to continue moving within the target region and accumulate rewards after reaching the goal.

   We report the average cumulative reward per trajectory across evaluation episodes. This metric jointly reflects the agent's ability to reach the target, remain within the target region once reached, and execute the sequence of pretrained skills reliably over long horizons, thereby evaluating the effectiveness of hierarchical skill composition without additional low-level fine-tuning.

E.4 Implementation Details

Our method is implemented based on the Soft Actor-Critic (SAC) backbone. We implement our method and baselines on top of the publicly available METRA codebase[4] (Park et al., 2024). Code is provided in the supplementary material. The key hyperparameters for the environment, state encoder, and RL optimization are listed in Table 4.

Trajectories in the unlabeled exploratory dataset are collected by running a standard METRA policy and selecting those that do not collapse to suboptimal behaviors. We select a small set of 32 trajectories for state-based tasks and 64 trajectories for pixel-based tasks. To illustrate the trajectory quality, we visualize the spatial distribution of these trajectories, as in Fig. 15. Importantly, CURE does not require a clean success-only dataset. As shown in Sec. 5.3 and Appendix D.2, replacing up to 50% of the pretraining data with failed trajectories causes only a minor performance degradation.

[4] https://github.com/seohongpark/METRA

Table 4: Hyperparameters for our experiments.

Hyperparameter	Value
State Encoder Architecture	
State Encoder Hidden Sizes	$[1024, 1024]$
Pixel Observation Encoder	CNN
Pixel Shape	64×64×3
State Encoder Pretrain Learning Rate	3×10^{-4} (Humanoid), 10^{-4} (Otherwise)
Dual Slack	0.001
Contrastive Learning Number of Negative Samples K	20
InfoNCE Temperature τ	0.2
SAC	
Actor/Critic Hidden Sizes	$[1024, 1024]$
Actor/Critic Learning Rate	10^{-4}
State Encoder Normal Learning Rate	6×10^{-4} (Humanoid), 10^{-4}(Otherwise)
State Encoder Slow Learning Rate	3×10^{-4} (Humanoid), 2×10^{-5} (Otherwise)
Discount Factor (γ)	0.99
Replay Buffer Size	3×10^5 (Quadruped), 10^6 (Otherwise)
Optimizer	Adam
episodes per epoch	8
gradient steps per epochs	200 (Quadruped), 50(Otherwise)
Latent Space Dimension ($\mathcal{Z}$)	2

Table 5: Computational burden of pretraining.

Env name	GPU model	GPU memory	GPU hours
Humanoid	A100-PCIE-40GB	688.0 MiB	0.611 h
Quadruped	A100-PCIE-40GB	12022.0 MiB	4.62 h

E.5 Details of the Instantiated Generalized Absorbing MDP

We detail the construction of the star-shaped chain MDP illustrated in Fig. 2(b), which is a minimal instantiation of the Generalized Absorbing MDP defined in Definition C.3. The agent starts at a center point and selects from K discrete directions at each step. The state space combines a progression depth $h \in \{0, \dots, H\}$, a direction index $k \in \{0, \dots, K-1\}$, and an absorbing flag $is_abs \in \{0, 1\}$. The Observation of the agent is defined as the tuple $(h\cos\theta_k, h\sin\theta_k, is_abs)$, where $\theta_k = 2\pi k/K$. Each forward-progressing action advances the agent along its current direction; any other action triggers an immediate, irreversible fall into the co-located absorbing state. Formally, from a non-absorbing state $(h, k, 0)$ and its observation $(h\cos\theta_k, h\sin\theta_k, 0)$ with $h > 0$, the action $a = k$ advances the agent to $(h+1, k, 0)$, while any other action $a \neq k$ immediately forces an irreversible transition into the co-located absorbing state $(h, k, 1)$. From the origin state $(0, 0, 0)$, every action a is safe and leads to the first progressing state $(1, a, 0)$. Once any absorbing state is entered, all subsequent actions leave the state unchanged. This construction directly instantiates the failure-prone setting analyzed in Section 3.2: a uniform random policy succeeds in progressing with probability $1/K$ at each non-origin step, which aligns with the condition of Proposition C.1.

We implement tabular versions of METRA, LSD, and our method, CURE, within this environment. The state encoder ϕ is instantiated as a learnable d-dimensional vector per discrete state, stored in a PyTorch parameter dictionary. At each training round, the agent collects N trajectories using the current Q-table; the first round relies on uniform random actions, while subsequent rounds follow a greedy policy with respect to the Q-values for each skill latent vector z. For METRA and LSD, the encoder is updated via stochastic Lagrangian optimization: the inner product $\Delta\phi^\top z$ is maximized subject to $\|\Delta\phi\|_2 \leq 1$ (METRA) or $\|\Delta\phi\|_2 \leq \|s' - s\|_2$ (LSD). After the encoder update, the Q-function is refined by tabular value iteration using the intrinsic reward $r = (\phi(s') - \phi(s))^\top z$. For CURE, the encoder is first pretrained on K forward progressing trajectories using an InfoNCE loss with an added Lipschitz penalty, and then fine-tuned during the main skill-discovery phase.

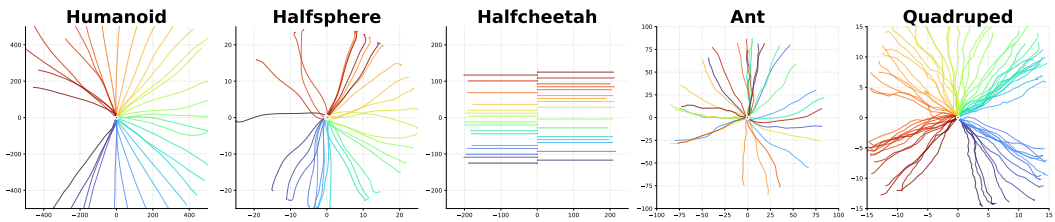


Figure 15: Visualizations of the exploratory trajectories in x-y space.

All methods share the same set of skill vectors z uniformly distributed on the unit circle, and are evaluated by the average depth reached and the total number of unique states covered.

E.6 Computational Resource Required

All experiments were conducted on a single workstation running Ubuntu 22.04 LTS, equipped with 440GB of memory and 8 NVIDIA GeForce RTX 4090 GPUs. The computational cost differs between the pretraining phase and the main skill discovery phase.

Pretraining. To evaluate the computational overhead, we benchmarked the GPU resource usage during pretraining on the two most computationally demanding environments (Humanoid and Quadruped). Detailed GPU consumption metrics are reported in Table 5.

Skill discovery. For most environments, training a full set of skills finishes within 2 days. The Humanoid environment is more expensive: because of its long episode horizon and the large number of training epochs required for the humanoid agent to acquire stable locomotion behaviors, one full training run typically requires 5 to 7 days.

F Further Discussions

F.1 Discussion on Absorbing Sets

In this paper, we use absorbing sets to model states where effective exploration no longer continues. Although in idealized reversible simulations, some failure states may not be strictly irreversible, recovery can be very unlikely for an unsupervised agent without task-specific rewards or recovery guidance within a practical interaction horizon. For example, after a humanoid falls, unsupervised exploration alone rarely enables the agent to discover the coordinated behavior needed to stand up. Real-world physical systems can also encounter genuinely irreversible failures, such as hardware damage caused by a fall or collision. Therefore, the absorbing-set abstraction captures both physically irreversible failures and states that are effectively unrecoverable during unsupervised skill discovery. Our goal is to reduce the likelihood that skill discovery prematurely enters such states, so that the agent can continue exploring the state space and learning diverse behaviors.

F.2 Complementary Roles of Contrastive Learning and Skill Discovery

Contrastive learning and skill discovery both operate without external rewards, but they play complementary roles in unsupervised reinforcement learning. Contrastive pretraining learns a geometric representation from offline trajectories. As shown in Proposition 4.1, InfoNCE aligns the latent representation with dominant temporal PCA directions to distinguish progressing transitions from absorbing failure transitions. However, contrastive learning only learns representations and does not actively explore the environment or obtain executable policies. In contrast, skill discovery trains a skill-conditioned policy to explore the state space and acquire diverse behaviors. Standard skill discovery trained from scratch can suffer from suboptimal entrapment because early failure transitions distort the learned representation and intrinsic rewards. CURE addresses this issue by using the contrastively pretrained encoder to initialize skill discovery and guide intrinsic reward learning. This geometric prior reduces the influence of early failure transitions and helps the policy continue exploring, while skill discovery utilizes this prior to learn diverse policies through active exploration. Thus, both components play unique and complementary roles to ensure robust and effective unsupervised skill learning.

F.3 Connection between CURE and Laplacian representations & eigenoptions.

Our geometric prior shares a theoretical foundation with Laplacian representations and eigenoptions (Machado et al., 2017), as both leverage transition dynamics to structure the behavior space. Eigenoptions perform spectral analysis on state transition graphs to extract principal axes of state diffusion (Machado et al., 2017). Similarly, contrastive pretraining via InfoNCE captures related spectral properties by identifying principal directions of transition displacement.

While CURE and eigenoptions share this geometric intuition, they are designed for different problem settings and downstream objectives, so their performance is not directly comparable. Below, we discuss their conceptual and methodological differences. First, classical eigenoptions (Machado et al., 2017) construct options directly to enhance exploration, whereas CURE uses the learned geometry to decouple representation learning from policy optimization. This decoupling breaks the positive feedback loop between representation updates and policy learning, preventing entrapment in suboptimal behaviors. Second, explicitly constructing graph Laplacians is computationally expensive and difficult to scale in high-dimensional continuous control. CURE bypasses graph construction by learning transition geometry directly from raw trajectory segments through batch contrastive learning. Finally, while eigenoptions define discrete macro-actions directly from eigenvectors, CURE provides a regularized prior for learning continuous latent skills. This prior enables the encoder and policy to jointly refine diverse continuous behaviors during downstream skill discovery.

F.4 Limitations and Future Works

First, our method requires a pretraining dataset that contains forward-progressing state transitions. Although Section 5.3 shows that CURE is robust to lower data quality, we have not quantified the minimum data quality required to ensure performance. Future work will investigate how pretraining data properties affect downstream performance, providing practical data collection guidelines.

Second, CURE currently uses an offline two-stage setup that pretrains the encoder before policy learning. While this decoupled design is computationally efficient, as shown in Table 5, it cannot use online interaction data to refine representations during skill learning. In future work, we plan to explore a continual learning framework that updates the encoder using new rollouts to adapt to newly explored states.

F.5 Discussion on Broader Impacts

This work has the potential to improve the safety and reliability of autonomous agents by reducing irreversible failure during unsupervised training. The proposed method can lead to lower training costs and energy consumption in real-world robotic applications by minimizing the need for human intervention and manual resets. Also, the proposed method is compatible with existing skill discovery frameworks as a lightweight plugin, which may strengthen the robustness of downstream task learning and simplify the design of exploration strategies.