A Unified Framework for Unsupervised Reinforcement Learning Algorithms

Siddhant Agarwal[1*], **Caleb Chuck**[2*], **Harshit Sikchi**[1*], **Jiaheng Hu**[1], **Max Rudolph**[1], **Scott Niekum**[3†], **Peter Stone**[1, 4†], **Amy Zhang**[1†]

[1]**The University of Texas at Austin** [2]**Synthefy** [3]**University of Massachusetts, Amherst** [4]**Sony AI**

[*] denotes equal contribution, [†] denotes equal advising

Abstract

Many sequential decision-making domains, from robotics to language agents, are naturally multi-task on the same set of underlying dynamics. Rather than learning each task separately, unsupervised reinforcement learning (RL) algorithms pretrain without reward, then leverage that pretraining to quickly obtain optimal policies for complex tasks. To this end, a wide range of algorithms have been proposed to explicitly or implicitly pretrain a representation that facilitates quickly solving some class of downstream RL problems. Examples include Goal-conditioned RL (GCRL), Mutual Information Skill Learning (MISL), forward-backward representation learning (FB) and controllability representations. This paper brings together all these heretofore distinct algorithmic frameworks into a unified view. First, we show that these algorithms are, in fact, approximating the same intractable representation learning objective, the successor measure or discounted future policy-dependent state-action distribution, under different assumptions. We then illustrate that to make these methods tractable, practical applications of these algorithms utilize embeddings that can be described under the framework of state equivalences. Through this work, we highlight shared underlying properties that characterize core problems in Unsupervised RL.

1 Introduction

Reinforcement Learning (RL) algorithms learn complex policies by identifying the complex interplay between actions, dynamics, and reward through trial-and-error. While RL has seen tremendous success across different fields (Chervonyi et al., 2025; Degrave et al., 2022; Wurman et al., 2022; Guo et al., 2025; Silver et al., 2017; Fawzi et al., 2022), it still relies on using a large number of environment interactions to learn a policy, which can make it prohibitively expensive. In many settings, such as robotics, the agent needs to solve a variety of tasks, described by different reward functions, in an single environment. Learning a new policy for for each new task can become prohibitively expensive. Consequently, Unsupervised RL offers a suite of techniques to first pretrain some useful characterization of the environment so that a wide variety of optimal policies can be inferred efficiently for a new, given task.

Over the years, many URL objectives Ma et al. (2022b); Touati et al. (2023); Agarwal et al. (2025); Barreto et al. (2017); Wang et al. (2024); Hu et al. (2024); Gregor et al. (2016); Machado et al. (2017a); Laskin et al. (2021) have been proposed for pretraining in the reward-free setting. Through these objectives, structures as varied as state encoders Rudolph et al. (2024), latent skills Eysenbach et al. (2022a), successor representations Dayan (1993), or goal-conditioned policies Agarwal et al. (2023) can be pretrained, and then applied for rapid downstream policy inference. On the surface, these techniques appear to be optimizing very different objectives, though with the same goal of rapid policy inference. With the proliferation of complex techniques, it can be challenging for researchers trying to apply URL to new contexts or improve upon URL techniques.

This work investigates a core question: Can all of these conceptually disparate methods be unified as variations of a single core algorithmic framework? At first glance, this may seem unlikely—these methods have significantly different loss objectives, from state coverage to goal-conditioned rewards, and learn different structures, from state representations to policies, each based on different intuitions and assumptions. However, recent work has established several bridges between different clusters of concepts, from successor measures to representation learning (Agarwal et al., 2025; Touati & Ollivier, 2021), or goal-conditioned RL to variational skills and empowerment (Choi et al., 2021). This paper aims to unify these seemingly distinct methods in two ways. First, we claim that each objective can be traced back to the core description of future policy-dependent state reachability, or the *successor measure*. Second, we observe a shared structure that all these algorithms use to make the successor measure tractable: *state feature equivalence under the successor measure*. Intuitively, we hypothesize that these methods tractably learn how the distribution of future states is affected by the policy (successor measure) by treating states with similar properties as equivalent (state feature equivalence).

While we do not claim to entirely cover the myriad of Unsupervised RL techniques, in this work our core contribution is to illustrate that this unified objective and structure exists in Goal-Conditioned Value Functions (GCVF) (Ma et al., 2022b), Mutual Information Skill Discovery (MISL) (Zheng et al., 2025; Eysenbach et al., 2022a), Proto-Successor Measures Agarwal et al. (2025), Proto-value Functions (Mahadevan, 2005), Successor Features (Dayan, 1993; Barreto et al., 2017) and Controllable Representations (Islam et al., 2023a; Rudolph et al., 2024). Intuitively, these concepts can be linked by simply recognizing that in order to pretrain a model that can be leveraged to get a policy for any reward function, these methods must learn some quantity or structure over the environment that effectively captures the relationship between state transitions and action sequences. In GCVF or MISL, this happens through policy-derived structures; in proto-successor measures and functions; and in successor features through learning linear value functions; and Controllable Representations use state embeddings. In this work we formalize the growing body of evidence Choi et al. (2021); Levy et al. (2023); Zheng et al. (2025); Fujimoto et al. (2025) showing that since these methods learn to characterize the same information (linking actions and dynamics) to achieve the same outcome (rapid policy inference given a reward) they are in fact fundamentally linked.

Our core contribution is twofold. First, we describe each of the aforementioned methods using a shared notation and demonstrate how their learning objectives can be framed as representing the successor measure. Second, we identify that to learn tractable, concise representations for successor measures, each method learns a suitable state abstraction implicitly or explicitly through a unified concept of state equivalences. To summarize, in this paper, we (1) draw connections between the unsupervised RL methods that utilize future predictability for efficient policy inference; (2) identify the unified objective that all of the different methods strive for, deriving how each method can be framed as an optimization of this unified objective; (3) identify the assumptions and approximations made by the various methods to solve the unified objective; and (4) relate the state abstractions learned by these methods through the perspective of state equivalences.

2 Preliminaries

All of the algorithms considered are assumed to operate on Markov Decision Processes (MDPs) (Puterman, 2014). A Markov Decision Process is a stochastic process defined as $(\mathcal{S}, \mathcal{A}, P, r, \gamma)$ where $\mathcal{S}$ denotes the set of states; $\mathcal{A}$ denotes the set of actions; $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the transition probability function, where $P(s' \mid s, a)$ is the probability of transitioning to state s' from state s after taking action a; $r : \mathcal{S} \to \mathbb{R}$ is the reward function; and γ is the discount factor. A policy, $\pi : S \to \Delta(A)$ is a function that outputs a distribution of actions for every state. The optimal policy for the MDP is defined to be the one maximizing the expected return: $J(\pi) = \mathbb{E}_\pi[\sum_t \gamma^t r(s_t)]$.

Additionally, we will be using the construct of reward-free MDPs (also mentioned in prior work Touati et al. (2023); Agarwal et al. (2025)) that are defined as $(\mathcal{S}, \mathcal{A}, P, \gamma)$. Any dynamical system can be approximated using a reward-free MDP. Infinitely many reward functions can be designed for a reward-free MDP. In other words, infinitely many MDPs can be constructed from a reward-free MDP.

Successor Measures: Successor Measures will play an important role in unifying the URL methods. Mathematically, successor measures define the measure over future states visited as M^π,

$$M^\pi(s, a, X) = \mathbb{E}_\pi[\sum_{t\geq 0} \gamma^t p^\pi(s_{t+1} \in X|s, a)] \;\; \forall X \subset \mathcal{S}. \tag{1}$$

Intuitively, they represent the discounted measure of ending up in a state $s^+ \in X$ starting from states s, taking an action a, and following the policy π thereafter. The most common form of successor measure used is $M^\pi(s, a, s^+)$ i.e the discounted measure of ending in the state s^+. Other common forms include $M^\pi(s, a, s^+, a^+)$ and $M^\pi(s, s^+)$.

Mutual Information: Mutual information computes the channel capacity between two random variables. For entropy denoted with H and KL divergence denoted as $D_{KL}(\cdot\|\cdot)$, mutual information between random variables A, B is defined as:

$$I(A; B) \coloneqq H(A) - H(A|B) = H(B) - H(B|A) \tag{2}$$

$$= D_{KL}(\mathbb{P}(A, B)\|\mathbb{P}(A)\mathbb{P}(B)) \tag{3}$$

Diverse Class of URL Algorithms: We will be focusing on Goal Conditioned RL, Mutual Information Skill Discovery, Successor Features, Proto-Successor Measures, Proto-Value Functions and Controllable Representations. Detailed background and related work on each have been proved in the supplementary material.

3 Successor Measure as a Unifying Objective

Each URL objective learns a different representation for MDP to allow for downstream policy inference. This raises the question: how do we reason about the commonality across these representations. In this section, we will argue that viewing these methods from the perspective of **Successor Measures** (M^π) estimation ties them together, bringing clarity to efficient downstream policy optimization. These methods either explicitly learn a compressed representation of successor measures or optimize a representation that allows them to implicitly use successor measures efficiently during inference. To illustrate this, we first introduce the unifying objective using successor measures. We will show that the proposed unified objective not only combines these different URL objectives, but also forms the basis for self-supervised representation learning in RL aimed for fast policy inference for any reward function. Because this objective is intractable, we will next provide a tractable approximation that will lead into the different URL objectives. In Section 4, we will discuss how a number of existing URL objectives stem from this approximation with different assumptions and present their tradeoffs.

3.1 The Unified Objective

The policy optimization for any reward function can be rewritten using successor measures (Kemeny et al., 1969; Touati & Ollivier, 2021; Agarwal et al., 2025):

$$\pi^* = \arg\max_\pi \sum_{s^+} M^\pi(s, a, s^+) r(s^+). \tag{4}$$

This policy inference clearly indicates why successor measures form such a crucial element in URL algorithms – they provide reward-independent representations and a linear objective for policy optimization. This implies that our representations are not tied to a set of predefined tasks and that the policy optimization step is computationally efficient as a function of these representations. Our proposed algorithmic framework can be divided into two phases, the **Pretraining** or **Representation Learning** phase and the **Policy Inference** phase.

The Pretraining Phase uses task-agnostic environment interactions to learn representations suitable for policy inference. Thus, this phase investigates the question: *how can we frontload computation for policy optimization to the pretraining phase if we don't have access to reward functions?* Successor Measures provide the answer to this question due to two key traits: 1) they are reward-free representations that can convert policy optimization into a linear objective, and 2) they characterize the notion of predicting the future distribution of an agent for any policy, which can be seen as the controllability of the agent. Then during the policy inference stage, the pretrained representation of mapping from policies to corresponding induced successor measure can be utilized to provide a near-optimal policy efficiently for any given reward function. In practice, based on assumptions about the distribution over downstream tasks/rewards and varying assumptions about the policy inference stage, prior URL algorithms suggest seemingly different pretraining objectives. Our proposed unified objective for unsupervised RL that ties in a broad class of prior methods can be denoted as follows:

Box 3.1: Unified Objective

Pretraining Phase

$$\textit{Learn: } M^\pi(s,a,s^+) \qquad \forall s \in \mathcal{S} \qquad \forall a \in \mathcal{A} \qquad \forall s^+ \in \mathcal{S} \qquad \forall \pi \in \Pi \tag{5}$$

Policy Inference Phase :

$$\textit{For a reward } r, \pi^* = \arg\max_{\pi \in \Pi} \sum_{s^+} M^\pi(s,a,s^+) r(s^+) \tag{6}$$

Proposition 3.1. *The algorithm presented in the Algorithm Box 3.1 is sufficient to produce optimal policies for any reward function.*

The unified objective is simple: Learn successor measures for any policy (Π represents the class of all possible policies in the MDP), for any state-action pair. Then policy inference is simply a search using the linear product of successor measure and reward, as seen in Equation 6. However, while simple this objective is still intractable.

The main reason for why the objective is intractable is that there is no way to characterize the class of all possible policies: Π. There can be $|\mathcal{A}|^{|\mathcal{S}|}$ possible deterministic policies in an MDP with finite state and action spaces, and this number can be infinite for MDPs with infinite (or continuous) states or actions. This makes characterizing a mapping from policy to the corresponding successor measures intractable. How can we perform an efficient search for $\pi \in \Pi$ during the policy inference phase from such a large non-parametric set? We introduce a tractable approximation in the next section, which we will show has connections to the different prior URL algorithms.

3.2 A Tractable Approximation

The intractability of the unified objective comes from the large non-parametric class of policies Π. Different URL methods approximate this policy class using a parametric approximation of the policy class using latent representation z. Mathematically, $\Pi := \{\pi_z | z \in \mathcal{Z}\}$ with $\pi \in \Pi$ being reduced to $z \in \mathcal{Z}$. This parametric set of policies $\mathcal{Z}$ is interpreted differently for different algorithms: these could be the set of goals (Kaelbling, 1993), set of skills (Eysenbach et al., 2018a), a set of possible linear weights for the reward span (Touati & Ollivier, 2021), or a discrete codebook (Agarwal et al., 2025). Thus $\mathcal{Z}$ defines the class of policies for which successor measure is represented. Additionally, define $\mathcal{T}$ which is the set of reward functions for which the policy inference will be valid. Ideally the $\mathcal{T}$ should be the set of all reward functions but based on the approximations and assumptions on the representation space of M^π and the space of policies Π. Due to these approximations, it may be possible that during policy inference searches over a policy space that is different from Π.

4 Unsupervised RL Objectives as Special Cases

In this section, we pose each of the URL objectives within the same framework of the single, unified objective. We will highlight the assumptions and compressions learned by each to produce corresponding tractable objectives that are widely used today. We will show that each of these objectives learns to represent a compact approximation of the successor measure implicitly or explicitly. These methods use this representation to either directly optimize Equation 6 or produce samples from M^π to optimize the expectation $\mathbb{E}_{M^\pi}[r]$. We will introduce a number of cross equivalences as well that deeply connect these objectives with one another, further establishing the unification. These different methods are compared against each other based on: 1) the distribution of tasks/rewards ($\mathcal{T}$) for which they produce optimal or near-optimal policies, 2) their assumptions about the class of policy space (the latent z), and 3) the efficiency of their policy inference phase. The result of these equivalences is summarized in Table 1. All proofs for the theorems are included in the supplementary material.

4.1 Goal-conditioned Reinforcement Learning (GCRL)

Goal-conditioned RL optimizes for a policy (and a value function) that is conditioned on the goal state $z \in \mathcal{S}$ that the agent has to reach. Mathematically, GCRL is expected to produce $V^*(s,z) = \max \mathbb{E}_\pi[\sum_t \gamma^t r_z(s_t,a_t)|s]$ (or $Q^*(s,a,z)$) where $r_z(s_t,a_t) = (1-\gamma)p(s_{t+1} = z|s_t,a_t)$ otherwise. In its most expansive sense, the goal set is the same as the set of states with GCRL being capable of producing the value of any state conditioned on any state in the MDP.

Under the lens of Unification: The equivalences between GCRL and Successor measures have already been hinted at in contrastive RL Eysenbach et al. (2021) where GCRL was seen as a density estimation problem. We extend this formally here with the following assumptions.

Assumption 4.1 (GCRL Policy Assumption)**.** Let $\mathcal{Z} \subseteq \mathcal{S}$ with $\Pi = \{\pi_z | z \in \mathcal{S}$ and $\pi_z$is optimal policy to reach goal $z\}$.

This assumption formally defines the tractable class of policies that is considered by GCRL. Consider the next assumption on the set of tasks or rewards for which GCRL performs policy inference,

Assumption 4.2 (GCRL Reward Assumption)**.** The set of rewards $\mathcal{T}$ is given by $\mathcal{T} = \{(1-\gamma)p(s_{t+1} = z|s_t,a_t) \quad \forall z \in \mathcal{Z}\}$.

With the assumptions formally defined for GCRL, we can bring GCRL into the unified objective:

Theorem 4.3. *With Π and $\mathcal{T}$ defined as per Assumptions 4.1 and 4.2, GCRL learns $Q^{\pi_z}(s,a) \propto M^{\pi_z}(s,a,z)$ for $s \in \mathcal{S}, z \in \mathcal{Z}, a \in \mathcal{A}$. The optimal policy inference for reward, r_z is π_z by construction.*

Additional Equivalences Approaches such as VIP (Ma et al., 2022b) and HILP (Park et al., 2024) additionally parameterize M^{π_z} as a metric ($M^{\pi_z} \propto -\|\phi(s) - \phi(z)\|$) to provide an inductive bias for representation learning. Similarly, contrastive RL Eysenbach et al. (2022b) approaches consider a low-rank parameterization ($M^{\pi_z} \propto \psi(s,a)^\top \phi(z)$) of M^π.

4.2 Mutual Information Skill Learning (MISL)

MISL objectives have been primarily used to discover skills-conditioned policies, where the skills are represented using a latent variable Z. While MISL approaches have large variation in their overall algorithms, the core has always been to maximize the mutual information between states and "skills" ($I(S;Z)$) or between transitions and skills ($I(S,S';Z)$). The details of the optimization can be found in the supplementary. Since computing the mutual information exactly is intractable, MISL methods often rely on lower bounds that require training a variational distribution $q(z|s)$ (or $q(z|s,s')$) representing posterior distribution of skills which defines the reward for policy optimization conditioned on z.

Under the lens of unification We demonstrate that variational distribution $q(z|s)$ can be used to estimate successor measures (Theorem 4.6). The policy class Π is not generally fixed in MISL, but rather emerges as a property of the objective. At convergence, the following assumption holds,

Assumption 4.4 (MISL Policy Assumption)**.** $\mathcal{Z}$, the set of diverse skills recovered by MISL, i.e $\Pi = \{\pi_z | z \in \mathcal{Z}$ i.e. π_z is a skill discovered by MISL $\}$ is sufficient to cover Π.

The set of skills discovered by MISL algorithms can be discrete (Eysenbach et al., 2018a; 2022a) or continuous (Park et al., 2023c; Zheng et al., 2025). Eysenbach et al. (2022a) makes an interesting finding that $\mathcal{Z}$ represents the set of policies optimal for some reward function and in general MISL does not recover all optimal policies.

We can define the assumption on the set of tasks considered by MISL,

Assumption 4.5 (MISL Reward Assumption)**.** The set of rewards $\mathcal{T} = \{r \mid \exists\ z \in \mathcal{Z}$ s.t. $\pi_z \in \arg\max_\pi \mathbb{E}_\pi[\sum_t \gamma^t r_t]\}$.

Finally, we can connect MISL to the unified objective using Theorem 4.6:

Theorem 4.6. *For Π defined using Assumption 4.4 and $\mathcal{T}$ defined using Assumption 4.5, MISL objectives learn $M^{\pi_z}(s, s^+) = \frac{q(z|s^+,s)p(s^+|s)}{p(z)}$ for $s \in \mu$, $a \sim \pi_z(\cdot|s \sim \mu)$ and $s^+ \in \mathcal{S}$. The policy inference can be performed by searching through the space of $z \in \mathcal{Z}$ for rewards defined in $\mathcal{T}$.*

The policy inference step in the above theorem is not as simple as described, as the set of rewards $\mathcal{T}$ is not known. Prior work has used hierarchical policy inference (Eysenbach et al., 2018a) and warm starting their policy networks (Eysenbach et al., 2018a) or exploration buffers (Eysenbach et al., 2022a).

Additional Equivalences Recent work (Zheng et al., 2025) leverages the relationship between MISL objective and InfoNCE as a variational lower bound Poole et al. (2019b). An unnormalized variational lower bound can be derived for the mutual information as follows,

Theorem 4.7. *(Zheng et al., 2025) Given a critic function, $f : \mathcal{S} \times \mathcal{S} \times \mathcal{Z} \to \mathbb{R}$, $I^\pi(S, S'; Z) \geq \mathbb{E}_{p^\pi(s,s',z)}[f(s, s', z)] - \mathbb{E}_{p^\pi(s,s')}[\log \mathbb{E}_{p(z)}[e^{f(s,s',z)}]]$ where the right hand side is the variational lower bound: $(VLB(f, \pi))$*

Theorem 4.7 opens wide connections between MISL and Contrastive RL approaches based on InfoNCE objectives like (Zheng et al., 2023; Myers et al., 2024). These connections have been utilized by Zheng et al. (2025); Park et al. (2023c) to extract state-representations from MISL which are different from the traditional variational compression from $q(z|s)$ or $q(z|s, s')$.

The relationship between GCRL and MISL has been studied by prior work through the lens of variational empowerment (Choi et al., 2021). Each diverse skill, z, is perceived to be a goal-conditioned policy π_z (policy conditioned to reach the goal z). More formally,

Theorem 4.8. *(Choi et al., 2021) For $\mathcal{Z} = \mathcal{S}$, GCRL with $r(s|z) = -\frac{1}{\sigma^2}\|z - s\|$ is the same as solving the MISL objective with the variational distribution, $q(z|s) = \mathcal{N}(z - s, \sigma^2)$.*

4.3 Successor Features (SF)

A number of prior approaches (Dayan, 1993; Barreto et al., 2017) consider a set of reward functions that are spanned by basis features (often denoted by ϕ) i.e. $\mathbf{r} = \Phi^\top w$ for some weight w. ϕ can depend on state, state-action or state-action-next state in the most general case, but for ease of exposition we restrict ourselves to state-features. For these methods, the cumulative state feature is called the successor feature, $\psi^\pi(s, a) = \mathbb{E}_\pi[\sum_t \gamma^t \phi(s_t)|s, a]$, and is used to define Q-functions (for reward $\Phi^\top w$) as $Q^\pi(s, a) = \psi^\pi(s, a)^\top w$. While several prior works (Barreto et al., 2017; Zhu et al., 2024) define the state features ϕ using fixed, random or Fourier features, others (Park et al., 2024; Agarwal et al., 2025) have specialized objectives that add different inductive biases to these features. There are a few methods (Touati & Ollivier, 2021; Filos et al., 2021) that have been able to jointly produce ϕ and ψ by optimizing for M^π.

Under the lens of unification The connections between successor features and successor measures has already been established in prior literature (Touati et al., 2023; Agarwal et al., 2025). Here, we situate prior works in the unified framework by first posing the assumption that follows from the definition of SF:

Assumption 4.9. The set of rewards $\mathcal{T}$ is assumed to be restricted to $\mathcal{T} = \{\mathbf{r} | \mathbf{r} = \Phi^\top z \text{ for some } z \in \mathbb{R}^d\}$.

To enable fast policy inference, a number of prior works assume an injective relationship between optimal policy and reward. Optimal policies are represented by the same latent that defines the reward function

Assumption 4.10. The set of policies is assumed to be restricted to $\Pi = \{\pi_z \mid \pi_z \text{ is optimal for the reward } \mathbf{r} = \Phi^\top z\}$.

This assumption has lead to wide success as policy inference simply boils down to linear regression to find the z that fits the reward function: $z^* = \arg\min_z [(\mathbf{r} - \Phi^\top z)^2]$. This assumption also leads to suboptimalities as discussed in (Sikchi et al., 2025).

With these assumptions, we can finally write the SF in terms of the unified objective,

Theorem 4.11. *With Π and $\mathcal{T}$ as defined by Assumptions 4.10 and 4.9, SF methods learn $M^{\pi_z}(s, a, s^+) = \psi(s, a, z)(\Phi^\top \Phi)^{-1}\Phi^\top, \forall\ s, s^+ \in \mathcal{S}$ and $a \in \mathcal{A}$. The inference on any reward function in $\mathcal{T}$ requires solving a linear regression problem, $z^* = \arg\min_z (r - \Phi^\top z)^2$.*

Additional equivalences The policy inference for SF involves solving a linear regression which also has a closed form solution. The Forward Backward representation (Touati & Ollivier, 2021) modifies SFs to further make the inference more efficient.

Theorem 4.12. *If the successor measure is parameterized as, $M^\pi(s, a, s+) = F(s, a, z)^\top B(s^+)$, with $B(s^+) = (\Phi^\top \Phi)^{-1}\phi^\top(s^+)$ and $F(s, a, z) = \psi(s, a, z)$, the algorithm in Theorem 4.11 reduces to the FB algorithm (Touati & Ollivier, 2021). The policy inference simply becomes $z^* = Br$.*

Several SF works have been designed that have connected other forms of URL like GCRL and MISL. For instance, HILP (Park et al., 2024) uses state-features learned to be sufficient to represent goal-reaching value functions:

Theorem 4.13. *If $\phi = \arg\min_\phi \mathbb{E}_{s,s',g}[\ell_\tau(||\phi(s) - \phi(g)|| - \mathbb{1}_{s \neq g} - \gamma||\phi(s') - \phi(g)||)]$ in Theorem 4.11, with $r(s, s', z) = (\phi(s) - \phi(s'))^\top z$, the resulting algorithm is HILP (Park et al., 2024).*

A similar connection can be drawn to recent MISL works. CSF (Zheng et al., 2025) uses an InfoNCE lower bound for the mutual information objective to learn state features which are then used to learn successor features. With Successor Features, policy inference is more efficient compared to other MISL approaches.

Theorem 4.14. *If $\phi = \arg\max_\phi \mathbb{E}_{p^\pi(s,s',z)}[(\phi(s) - \phi(s'))^\top z] - \mathbb{E}_{p^\pi(s,s')}[\log \mathbb{E}_{p(z)}[e^{(\phi(s)-\phi(s'))^\top z}]]$, in Theorem 4.11, with $r(s, s', z) = (\phi(s) - \phi(s'))^\top z$, the resulting algorithm is CSF (Zheng et al., 2025).*

4.4 Proto Successor Measures (PSM)

Proto Successor Measure (PSM) (Agarwal et al., 2025) uses the linearity of the Bellman equations to define a decomposition of successor measure using basis vectors, $M^\pi = \phi w^\pi + b$. This parameterization makes PSM similar to successor features but the representation is simpler as ϕ is independent of policy π.

Under the lens of Unification PSM directly learns a representation for M^π and uses these representations to infer a policy for any reward function. PSM uses a discrete codebook $z \in \mathbb{I}^+$ to parameterize the distribution of policies. The policy π_z is given by $\text{Uniform}(z + hash(obs))$. Formally the approximation is as follows,

Assumption 4.15 (PSM Policy Assumption)**.** The set of policies Π is approximated as, $\Pi = \{\pi_z \mid \pi_z = Uniform(z + hash(obs)), z \in [0, 2^h] \cap \mathbb{I}\}$.

PSM does not make any assumptions on the reward class and hence can produce optimal policies for $\mathcal{T} = \{$ All reward functions $\}$. The inference step requires solving a constrained linear program $\arg\max_w \phi w r$ s.t. $\phi w + b \geq 0$.

Theorem 4.16. *PSM learns $M^{\pi_z}(s, a, s^+) = \sum_i \phi_i(s, a, s^+) w_i^{\pi_z} + b(s, a, s^+)$ for $\pi_z \in \Pi$ as defined in Assumption 4.15.*

Additional Equivalences PSM has pretty strong connections to Successor Features. Agarwal et al. (2025) had introduced the theorem,

Theorem 4.17. *For the PSM representation $M^\pi(s, a, s^+) = \phi(s, a, s^+) w^\pi + b(s, a, s^+)$ and $\phi(s, a, s^+) = \phi_\psi(s, a)^T \varphi(s^+)$, the successor feature $\psi^\pi(s, a) = \phi_\psi(s, a) w^\pi$ for the state feature $\varphi(s)^T (\mathbb{E}_\rho(\varphi\varphi^T))^{-1}$.*

4.5 Proto Value Functions (PVF)

Proto Value Functions (Mahadevan & Maggioni, 2007) decompose the value function into a spectral basis, $V^\pi(s) = \phi(s)^\top w^\pi$ or $Q^\pi(s, a) = \phi(s, a)^\top w^\pi$. A number of works (Mahadevan, 2005; Farebrother et al., 2023) have extended this construction into several interesting settings. This representation looks similar to PSM, but here the value function undergoes a spectral decomposition rather than successor measures. The spectral basis has been obtained either directly using an eigen-decomposition of the graph-Laplacian (Mahadevan, 2005) or approximated as the mean error over fitting auxiliary value functions Farebrother et al. (2023); Bellemare et al. (2019).

Under the lens of unification Prior works Farebrother et al. (2023); Bellemare et al. (2019) have drawn connections between these representations and successor measures and the set of value functions represented by them.

Assumption 4.18 (PVF Policy Assumption)**.** The class of policy Π is assumed to be $\{\pi_U\}$ or a uniformly random policy.

The set of downstream tasks that can be solved by these methods is not trivial to define. Bellemare et al. (2019) describes how these spectral methods represent value functions belonging to the set $\mathcal{V} = \{V | V$ is in the convex hull of $V^{aux}\}$ where V^{aux} is the set of auxiliary value functions defined by the set $V^{aux} = \{(I - \gamma P^\pi)^{-1} r_z\}$ and r_z is an indicator reward $r_z = \mathbb{1}_{s=z}$. Formally, the assumption is as follows,

Assumption 4.19 (PVF Reward Assumption)**.** For $V^{aux} = \{(\mathbb{I} - \gamma P^\pi)^{-1} r_z\}$ and $\mathcal{V}$ be the ConvexHull(V^{aux}), the set of downstream rewards are assumed to be $\mathcal{T} = \{r \mid V^* \in \mathcal{V}\}$.

The following theorem connects PVF to the unified objective,

Theorem 4.20. *The eigenvectors used by PVFs are the same as that of $M^{\pi_U}(s, s^+)$. Therefore, PVFs learn $M^{\pi_U}(s, s^+) = \phi w$. The policy inference for a reward function in the class $\mathcal{T}$ follows from the LSPI algorithm.*

It has already been shown (Theorem 4.4 of Agarwal et al. (2025)) that PVFs learn a smaller class of optimal value functions than spectral decomposition of successor measures.

4.6 Controllable Representations

Controllable representation learning compresses the states to deal with only the controllable factors of the state. All of them learn state embeddings that identify what can be controlled in the state. Several prior approaches (Islam et al., 2023a; Lamb et al., 2022; Levine et al., 2024; Rudolph et al., 2024) have used inverse dynamics models, $p(a|s, s')$ to model controllability. These representations learn the minimum necessary state information to recover actions, but are often insufficient to measure long term controllability. Extending these representations to multi-step requires k-step inverse dynamics models (Islam et al., 2023a; Lamb et al., 2022; Levine et al., 2024) or recursive computations through Wasserstein distance (Rudolph et al., 2024).

Under the lens of unification These methods learn state abstractions that make them stand apart from all the other methods discussed here. But their adherence to the use of multi-step future predictability ties them back to the notion of successor measures. We start with the first assumption (4.21) that defines the setting of Exo-MDPs. The formal definition of Exo-MDPs can be found in (Efroni et al., 2022) and is also provided in the supplementary material.

Assumption 4.21 (Exo-MDPs)**.** It is possible to learn a mapping $\phi : \mathcal{S} \to \mathcal{X}$ with $|\mathcal{S}| > |\mathcal{X}|$ such that $\mathcal{X}$ contains all the *endogenous components*.

The inference steps of these methods also differ from those previously discussed as they do not explicitly model M^π. Rather, they use the state compression ϕ as a representation for downstream RL, which defines the reward functions:

Assumption 4.22. The set of rewards $\mathcal{T}$ considered is the set of all possible reward functions on the endogenous component $\mathcal{X}$.

These methods use a behavioral policy, π_β, to reason about multi-step controllability and learn using the successor measure based only on π_β, M^{π_β}. Methods such as Rudolph et al. (2024); Levine et al. (2024) use a uniform random policy as the behavioral policy.

Assumption 4.23. The set of policies for which M^π is learned (or implicitly estimated) is $\Pi = \{\pi_\beta\}$.

Methods by Lamb et al. (2022); Islam et al. (2023a); Levine et al. (2024) model $P(a_t|\phi(s_t), \phi(s_{t+k}))$ using a classifier f. They use the classifier to reason about $(s_t,\ s_{t+k})$ for $k \in [1, K]$. In some sense, the classifer f is trying to model $\sum_{k=1}^K P(a_t|s_t, s_{t+k})$ (in case of Islam et al. (2023a)) or $\sum_{k=1}^K P(a_t|s_t, s_{t+k}) = \sum_{k=1}^K f(\cdot\ ,\cdot\ , k)$ (in case of Lamb et al. (2022); Levine et al. (2024). Define M_K^π as the K-step undiscounted successor measure, $M_K^\pi(s, a, s^+) = \sum_{k=1}^K P(s_{t+k} = s^+|s_t, a_t)$. Consider the following theorem,

Theorem 4.24. *Multi-step inverse methods like Lamb et al. (2022); Islam et al. (2023a); Levine et al. (2024), model* $M_K^{\pi_\beta}$, $\ \forall\ s \in \mathcal{S},\ a \in \mathcal{A},\ s^+ \in \mathcal{S}$ *as* $M_K^{\pi_\beta}(s, a, s^+) = \frac{f(a|s,s^+)p^{\pi_\beta}(s^+|s)}{\pi_\beta(a|s)}$.

On the other hand, Action-Bisimulation (Rudolph et al., 2024) uses the recursive definition of bismulation metrics to reason about an infinite horizon multi-step controllability. It can be shown through Theorem 4.25 that the state compression obtained by Action-Bisimulation is a result of equivalences predicted using successor measures,

Theorem 4.25. *In Action-Bisimulation (Rudolph et al., 2024),* $||\phi(s_1) - \phi(s_2)|| = 0 \Leftrightarrow M^{\pi_U}(s_1, a, s^+) = M^{\pi_U}(s_2, a, s^+)$, $\forall\ a \in \mathcal{A},\ s^+ \in \mathcal{S}$ *where* π_U *is a uniformly random policy.*

Additional Equivalences Controllable representations focus more on learning state abstractions. We discuss the comparisons of state abstractions extracted from all the URL methods in the next section.

5 Tractable Objectives require State Abstractions

We introduced the algorithmic framework 3.1 which is intractable due to the enumeration of all policies being exponential in the states. We described in Section 4 how different algorithms represent successor measures for only a reduced class of policies. It is evident that there is a tradeoff in performance that depends on the size of Π. If a very large class of Π is represented, the policy inference search is more expensive; if the class of Π is very small, the representations are not informative enough and the optimal policy cannot be found.

Algorithm Class	M^π **Approximation**	**Policy Inference**	$d(\phi(s_1), \phi(s_2))$ **for State Equivalences**
GCRL	$Q^{\pi_z}(s,a) \propto M^{\pi_z}(s,a,z)$	Direct for $\mathcal{T} = \{r_z(s_t,a_t) = (1-\gamma)p(s_{t+1}=z\|s_t,a_t)\}$	$-\|\|\phi(s_1)-\phi(s_2)\|\|$
MISL	$M^{\pi_z}(s,s^+) = \frac{q(z\|s^+,s)p(s^+\|)}{p(z)}$	Search over $\mathcal{Z}$ for $\mathcal{T} = \{r \mid \pi^*(r) \in \{\pi_z\}\}$	$D_{\text{KL}}(q_\phi(z\|s_1) \parallel q_\phi(z\|s_2))$
SF	$M^{\pi_z}(s,a,s^+) = \psi(s,a,z)(\Phi^\top\Phi)^{-1}\Phi^\top$	Linear Regression for $\mathcal{T} = \{\mathbf{r}\|\mathbf{r} = \Phi^\top z$ for some $z \in \mathbb{R}^d\}$	$\phi(s_1)^\top\phi(s_2)$
PSM	$M^{\pi_z}(s,a,s^+) = \sum_i^d \phi_i(s,a,s^+)w_i^\pi + b(s,a,s^+)$	Constrained LP for $\mathcal{T} =$ Any reward	$\phi(s_1)^\top\phi(s_2)$
PVF	$M^{\pi_U}(s,s^+) = \phi w$	LSPI for $\mathcal{T} = \{$ Any r for which $V^* \in$ convex hull of $V^{aux}\}$	$\phi(s_1)^\top\phi(s_2)$
Controllable Rep.	$M_K^{\pi_\beta}(s,a,s^+) = \frac{f(z\|s,s^+)p(s^+\|s)}{\pi_\beta(a\|s)}$	Full RL with compressed state space	$-\|\|\phi(s_1)-\phi(s_2)\|\|$

Table 1: Comparison of Unsupervised Reinforcement Learning Methods

We argue that these methods implicitly or explicitly learn state abstractions that are suitable for planning and lead to a concise form of M^π. These abstractions define state equivalences in the MDP. Formally, consider $\phi : \mathcal{S} \to \mathcal{X}$ as a state abstraction. An ideal abstraction would have $\phi(s_1) = \phi(s_2) \iff s_1 = s_2$ but this implies no compression or $|\mathcal{X}| = |\mathcal{S}|$. In practical settings, we want an abstraction that preserves the future predictability s. In other words, ϕ should be such that $M^\pi(s,a,s^+) = M^\pi(\phi(s), a, \phi(s^+))$.

Using state abstractions, state equivalences in the compressed space can be shown to follow,

Definition 5.1. $\phi(s_1) = \phi(s_2)$ iff $M^\pi(\phi(s_1), a, \phi(s^+)) = M^\pi(\phi(s_2), a, \phi(s^+))$.

Finally, we can define how these different URL objectives implicitly (or explicitly) define these state abstractions. For some metric d, $d(\phi(s_1), \phi(s_2)) \propto p(s_1 = s_2)$. The probability $p(s_1 = s_2)$ denotes the probability of the two states being equivalent. The metric d is specific to the respective URL method and is mentioned in Table 1.

6 Conclusion

Unsupervised RL can help significantly mitigate the sample efficiency challenge of solving complex tasks at test time by pretraining models that are useful for downstream inference. While this promise has attracted substantial investigation to this problem setting, this interest has also proliferated a wide variety of disparate objectives. As researchers continue to build upon this body of techniques, it can be challenging to identify unexplored areas and discriminate between such variegated techniques. In this work we offer a unified framework to understand some of the most popular and dissimilar methods. We demonstrate that each of these methods can be traced back to optimizing a form of the successor measure, and that they apply state equivalence to compress the underlying complexities to make this learning tractable. Through the lens of this objective, we hope to excite the reader with connections between the objectives for different methods, and through the perspective of abstraction to suggest novel cross-pollination of techniques. We hope this work will inspire investigation into questions like: Can hindsight be applied to successor features? Should world models be trained with explicit successor features? How does the exploration term in variational skills apply to expanding the space of proto-value functions? While this work is just a sampling of initial connections, we expect many more will become evident through the analysis in this work.

References

Siddhant Agarwal, Ishan Durugkar, Peter Stone, and Amy Zhang. f-policy gradients: A general framework for goal-conditioned rl using f-divergences. *Advances in Neural Information Processing Systems*, 36:12100–12123, 2023.

Siddhant Agarwal, Harshit Sikchi, Peter Stone, and Amy Zhang. Proto successor measure: Representing the behavior space of an rl agent. In *Forty-second International Conference on Machine Learning*, 2025.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 5055–5065, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

David Barber and Felix Agakov. The im algorithm: a variational approach to information maximization. *Advances in neural information processing systems*, 16(320):201, 2004.

André Barreto, Will Dabney, Rémi Munos, Jonathan J Hunt, Tom Schaul, Hado P van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

Kate Baumli, David Warde-Farley, Steven Hansen, and Volodymyr Mnih. Relative variational intrinsic control. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 6732–6740, 2021.

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

Marc G. Bellemare, Will Dabney, Robert Dadashi, Adrien Ali Taiga, Pablo Samuel Castro, Nicolas Le Roux, Dale Schuurmans, Tor Lattimore, and Clare Lyle. A geometric perspective on optimal representations for reinforcement learning, 2019. URL https://arxiv.org/abs/1901.11530.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Víctor Campos, Alexander Trott, Caiming Xiong, Richard Socher, Xavier Giró-i Nieto, and Jordi Torres. Explore, discover and learn: Unsupervised discovery of state-covering skills. In *International conference on machine learning*, pp. 1317–1327. PMLR, 2020.

Yuri Chervonyi, Trieu H. Trinh, Miroslav Olšák, Xiaomeng Yang, Hoang Nguyen, Marcelo Menegali, Junehyuk Jung, Vikas Verma, Quoc V. Le, and Thang Luong. Gold-medalist performance in solving olympiad geometry with alphageometry2, 2025. URL https://arxiv.org/abs/2502.03544.

Jongwook Choi, Archit Sharma, Honglak Lee, Sergey Levine, and Shixiang Shane Gu. Variational empowerment as representation learning for goal-based reinforcement learning. *CoRR*, abs/2106.01404, 2021. URL https://arxiv.org/abs/2106.01404.

Caleb Chuck. *Control-based factorization through causal interactions and hierarchical reinforcement learning*. PhD thesis, 2024.

Caleb Chuck, Supawit Chockchowwat, and Scott Niekum. Hypothesis-driven skill discovery for hierarchical deep reinforcement learning. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5572–5579. IEEE, 2020.

Caleb Chuck, Kevin Black, Aditya Arjun, Yuke Zhu, and Scott Niekum. Granger-causal hierarchical skill discovery. *arXiv e-prints*, pp. arXiv–2306, 2023.

Caleb Chuck, Fan Feng, Carl Qi, Chang Shi, Siddhant Agarwal, Amy Zhang, and Scott Niekum. Null counterfactual factor interactions for goal-conditioned reinforcement learning. *arXiv preprint arXiv:2505.03172*, 2025.

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

Jonas Degrave, Federico Felici, Jonas Buchli, Michael Neunert, Brendan Tracey, Francesco Carpanese, Timo Ewalds, Roland Hafner, Abbas Abdolmaleki, Diego de las Casas, Craig Donner, Leslie Fritz, Cristian Galperti, Andrea Huber, James Keeling, Maria Tsimpoukelli, Jackie Kay, Antoine Merle, Jean-Marc Moret, Seb Noury, Federico Pesamosca, David Pfau, Olivier Sauter, Cristian Sommariva, Stefano Coda, Basil Duval, Ambrogio Fasoli, Pushmeet Kohli, Koray Kavukcuoglu, Demis Hassabis, and Martin Riedmiller. Magnetic control of tokamak plasmas through deep reinforcement learning. *Nature*, 602(7897):414–419, 2022. DOI: 10.1038/s41586-021-04301-9. URL https://doi.org/10.1038/s41586-021-04301-9.

Ishan Durugkar, Mauricio Tec, Scott Niekum, and Peter Stone. Adversarial intrinsic motivation for reinforcement learning. *Advances in Neural Information Processing Systems*, 34:8622–8636, 2021.

Yonathan Efroni, Dipendra Misra, Akshay Krishnamurthy, Alekh Agarwal, and John Langford. Provably filtering exogenous distractors using multistep inverse dynamics. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=RQLLzMCefQu.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. *CoRR*, abs/1802.06070, 2018a. URL http://arxiv.org/abs/1802.06070.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. *arXiv preprint arXiv:1802.06070*, 2018b.

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

Benjamin Eysenbach, Ruslan Salakhutdinov, and Sergey Levine. The information geometry of unsupervised reinforcement learning. In *International Conference on Learning Representations*, 2022a. URL https://openreview.net/forum?id=3wU2UX0voE.

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, 2022b.

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

Alhussein Fawzi, Matej Balog, Aja Huang, Thomas Hubert, Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Francisco J R Ruiz, Julian Schrittwieser, Grzegorz Swirszcz, et al. Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature*, 610(7930):47–53, 2022.

Norm Ferns, Prakash Panangaden, and Doina Precup. Bisimulation metrics for continuous markov decision processes. *SIAM Journal on Computing*, 40(6):1662–1714, 2011.

Angelos Filos, Clare Lyle, Yarin Gal, Sergey Levine, Natasha Jaques, and Gregory Farquhar. Psiphi-learning: Reinforcement learning with demonstrations using successor features and inverse temporal difference learning. In *International Conference on Machine Learning*, pp. 3305–3317. PMLR, 2021.

Scott Fujimoto, Pierluca D'Oro, Amy Zhang, Yuandong Tian, and Michael Rabbat. Towards general-purpose model-free reinforcement learning. *arXiv preprint arXiv:2501.16142*, 2025.

Seyed Kamyar Seyed Ghasemipour, Richard Zemel, and Shixiang Gu. A divergence minimization perspective on imitation learning methods. In *Conference on robot learning*, pp. 1259–1277. PMLR, 2020.

Dibya Ghosh, Abhishek Gupta, and Sergey Levine. Learning actionable representations with goal-conditioned policies. *arXiv preprint arXiv:1811.07819*, 2018.

Robert Givan, Thomas Dean, and Matthew Greig. Equivalence notions and model minimization in markov decision processes. *Artificial intelligence*, 147(1-2):163–223, 2003.

Karol Gregor, Danilo Jimenez Rezende, and Daan Wierstra. Variational intrinsic control. *arXiv preprint arXiv:1611.07507*, 2016.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Jiaheng Hu, Zizhao Wang, Peter Stone, and Roberto Martín-Martín. Disentangled unsupervised skill discovery for efficient hierarchical reinforcement learning. *Advances in Neural Information Processing Systems*, 37:76529–76552, 2024.

Riashat Islam, Manan Tomar, Alex Lamb, Yonathan Efroni, Hongyu Zang, Aniket Didolkar, Dipendra Misra, Xin Li, Harm Van Seijen, Remi Tachet des Combes, et al. Agent-controller representations: Principled offline rl with rich exogenous information. *International Conference on Learning Representations 2023*, 2023a.

Riashat Islam, Manan Tomar, Alex Lamb, Yonathan Efroni, Hongyu Zang, Aniket Didolkar, Dipendra Misra, Xin Li, Harm van Seijen, Remi Tachet des Combes, and John Langford. Agent-controller representations: Principled offline rl with rich exogenous information, 2023b. URL https://arxiv.org/abs/2211.00164.

Leslie Pack Kaelbling. Learning to achieve goals. In *IJCAI*, pp. 1094–1099. Citeseer, 1993.

Liyiming Ke, Sanjiban Choudhury, Matt Barnes, Wen Sun, Gilwoo Lee, and Siddhartha Srinivasa. Imitation learning as f-divergence minimization. In *Algorithmic Foundations of Robotics XIV: Proceedings of the Fourteenth Workshop on the Algorithmic Foundations of Robotics 14*, pp. 313–329. Springer, 2021.

John G Kemeny, J Laurie Snell, et al. *Finite markov chains*, volume 26. van Nostrand Princeton, NJ, 1969.

Alexander S Klyubin, Daniel Polani, and Chrystopher L Nehaniv. Empowerment: A universal agent-centric measure of control. In *2005 ieee congress on evolutionary computation*, volume 1, pp. 128–135. IEEE, 2005.

Alex Lamb, Riashat Islam, Yonathan Efroni, Aniket Didolkar, Dipendra Misra, Dylan Foster, Lekan Molu, Rajan Chari, Akshay Krishnamurthy, and John Langford. Guaranteed discovery of control-endogenous latent states with multi-step inverse models, 2022. URL https://arxiv.org/abs/2207.08229.

Michael Laskin, Denis Yarats, Hao Liu, Kimin Lee, Albert Zhan, Kevin Lu, Catherine Cang, Lerrel Pinto, and Pieter Abbeel. Urlb: Unsupervised reinforcement learning benchmark. *arXiv preprint arXiv:2110.15191*, 2021.

Michael Laskin, Hao Liu, Xue Bin Peng, Denis Yarats, Aravind Rajeswaran, and Pieter Abbeel. Cic: Contrastive intrinsic control for unsupervised skill discovery. *arXiv preprint arXiv:2202.00161*, 2022.

Alexander Levine, Peter Stone, and Amy Zhang. Multistep inverse is not all you need, 2024. URL https://arxiv.org/abs/2403.11940.

Andrew Levy, Sreehari Rammohan, Alessandro Allievi, Scott Niekum, and George Konidaris. Hierarchical empowerment: Towards tractable empowerment-based skill learning. *arXiv preprint arXiv:2307.02728*, 2023.

Bo Liu, Yihao Feng, Qiang Liu, and Peter Stone. Metric residual networks for sample efficient goal-conditioned reinforcement learning, 2023. URL https://arxiv.org/abs/2208.08133.

Jason Yecheng Ma, Jason Yan, Dinesh Jayaraman, and Osbert Bastani. Offline goal-conditioned reinforcement learning via f-advantage regression. *Advances in Neural Information Processing Systems*, 35:310–323, 2022a.

Yecheng Jason Ma, Shagun Sodhani, Dinesh Jayaraman, Osbert Bastani, Vikash Kumar, and Amy Zhang. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022b.

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, 2017a.

Marlos C. Machado, Clemens Rosenbaum, Xiaoxiao Guo, Miao Liu, Gerald Tesauro, and Murray Campbell. Eigenoption discovery through the deep successor representation. *CoRR*, abs/1710.11089, 2017b. URL http://arxiv.org/abs/1710.11089.

Sridhar Mahadevan. Proto-value functions: Developmental reinforcement learning. In *Proceedings of the 22nd international conference on Machine learning*, pp. 553–560, 2005.

Sridhar Mahadevan and Mauro Maggioni. Proto-value functions: A laplacian framework for learning representation and control in markov decision processes. *Journal of Machine Learning Research*, 8(10), 2007.

Vivek Myers, Chongyi Zheng, Anca Dragan, Sergey Levine, and Benjamin Eysenbach. Learning temporal distances: Contrastive successor features can provide a metric structure for decision-making. *arXiv preprint arXiv:2406.17098*, 2024.

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

Soroush Nasiriany, Vitchyr Pong, Steven Lin, and Sergey Levine. Planning with goal-conditioned policies. *Advances in neural information processing systems*, 32, 2019.

Andrew Y Ng, Stuart Russell, et al. Algorithms for inverse reinforcement learning. In *Icml*, volume 1, pp. 2, 2000.

Seohong Park, Dibya Ghosh, Benjamin Eysenbach, and Sergey Levine. Hiql: Offline goal-conditioned rl with latent states as actions. *Advances in Neural Information Processing Systems*, 36:34866–34891, 2023a.

Seohong Park, Kimin Lee, Youngwoon Lee, and Pieter Abbeel. Controllability-aware unsupervised skill discovery. *arXiv preprint arXiv:2302.05103*, 2023b.

Seohong Park, Oleh Rybkin, and Sergey Levine. Metra: Scalable unsupervised rl with metric-aware abstraction. In *The Twelfth International Conference on Learning Representations*, 2023c.

Seohong Park, Tobias Kreiman, and Sergey Levine. Foundation policies with hilbert representations, 2024. URL https://arxiv.org/abs/2402.15567.

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

Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5171–5180. PMLR, 09–15 Jun 2019a. URL https://proceedings.mlr.press/v97/poole19a.html.

Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International conference on machine learning*, pp. 5171–5180. PMLR, 2019b.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Max Rudolph, Caleb Chuck, Kevin Black, Misha Lvovsky, Scott Niekum, and Amy Zhang. Learning action-based representations using invariance. In *Reinforcement Learning Conference*, 2024.

Harshit Sikchi, Rohan Chitnis, Ahmed Touati, Alborz Geramifard, Amy Zhang, and Scott Niekum. Score models for offline goal-conditioned reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=oXjnwQLcTA.

Harshit Sikchi, Andrea Tirinzoni, Ahmed Touati, Yingchen Xu, Anssi Kanervisto, Scott Niekum, Amy Zhang, Alessandro Lazaric, and Matteo Pirotta. Fast adaptation with behavioral foundation models. In *Reinforcement Learning Conference*, 2025. URL https://openreview.net/forum?id=soeW8RGo1N.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy P. Lillicrap, Karen Simonyan, and Demis Hassabis. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *CoRR*, abs/1712.01815, 2017. URL http://arxiv.org/abs/1712.01815.

Kimberly L. Stachenfeld, Matthew M. Botvinick, and Samuel J. Gershman. Design principles of the hippocampal cognitive map. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper_files/paper/2014/file/6083b607d0b81940c0280e465c79f5d5-Paper.pdf.

Ahmed Touati and Yann Ollivier. Learning one representation to optimize all rewards. *Advances in Neural Information Processing Systems*, 34:13–23, 2021.

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

Tongzhou Wang, Antonio Torralba, Phillip Isola, and Amy Zhang. Optimal goal-reaching reinforcement learning via quasimetric learning, 2023. URL https://arxiv.org/abs/2304.01203.

Zizhao Wang, Jiaheng Hu, Caleb Chuck, Stephen Chen, Roberto Martín-Martín, Amy Zhang, Scott Niekum, and Peter Stone. Skild: Unsupervised skill discovery guided by factor interactions. *arXiv preprint arXiv:2410.18416*, 2024.

Peter R. Wurman, Samuel Barrett, Kenta Kawamoto, James MacGlashan, Kaushik Subramanian, Thomas J. Walsh, Roberto Capobianco, Alisa Devlic, Franziska Eckert, Florian Fuchs, Leilani Gilpin, Piyush Khandelwal, Varun Kompella, HaoChih Lin, Patrick MacAlpine, Declan Oller, Takuma Seno, Craig Sherstan, Michael D. Thomure, Houmehr Aghabozorgi, Leon Barrett, Rory Douglas, Dion Whitehead, Peter Dürr, Peter Stone, Michael Spranger, and Hiroaki Kitano. Outracing champion gran turismo drivers with deep reinforcement learning. *Nature*, 602(7896): 223–228, 2022. DOI: 10.1038/s41586-021-04357-7. URL https://doi.org/10.1038/s41586-021-04357-7.

Chongyi Zheng, Ruslan Salakhutdinov, and Benjamin Eysenbach. Contrastive difference predictive coding. *arXiv preprint arXiv:2310.20141*, 2023.

Chongyi Zheng, Jens Tuyls, Joanne Peng, and Benjamin Eysenbach. Can a misl fly? analysis and ingredients for mutual information skill learning. *arXiv preprint arXiv:2412.08021*, 2024.

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

Chuning Zhu, Xinqi Wang, Tyler Han, Simon S Du, and Abhishek Gupta. Distributional successor features enable zero-shot policy optimization. *arXiv preprint arXiv:2403.06328*, 2024.

Appendix

A A Deep Dive into Unsupervised RL methods

A.1 Goal Conditioned Reinforcement Learning

Goal Conditioned RL refers to the class of algorithms that learn policies to reach certain goal states $g \in G$ where the set of goals is a subset of the state space $G \subseteq \mathcal{S}$. GCRL is the simplest and most common type of multi-task RL algorithms where the class of reward functions considered are simply one-hots on the goal states (notated $\mathbb{1}(s = g)$). However, even in this case a wide variety of alternative reward functions can be derived based on this including: (1-, termination, probabilistic). In Eysenbach et al. (2021) the probabilistic representation most directly captures the future state density. However, other forms have similar properties under transformations or assumptions.

A diverse set of prior works have built on the GC-MDPs (Kaelbling, 1993) to produce a large class of GCRL algorithms both in the online (Andrychowicz et al., 2017; Durugkar et al., 2021; Agarwal et al., 2023; Chuck et al., 2025) and offline settings (Ma et al., 2022a; Sikchi et al., 2024). GCRL has been proposed as self-supervised learning for learning state-reaching value functions from sequential data (Ma et al., 2022b). Several methods (Park et al., 2023c;a) use goals to define skills and use these to construct zero-shot policies (Park et al., 2023a) or for exploration (Park et al., 2023c). Goal-reaching policies can also be used as the action space for high level policies in hierarchical policy learning (Park et al., 2023a; Chuck et al., 2020; 2023), and in factored settings (Chuck, 2024; Chuck et al., 2025), where $\mathcal{Z}$ is a subset of factors, dictated by a given function $\phi : \mathcal{S} \to Z$, that selects the goal factors. Because of their simplicity, goal conditioned policies have also been applied to real world visual tasks with impressive succcess (Nair et al., 2018; Nasiriany et al., 2019).

Under the lens of unification, this diverse set of applications leverage certain assumptions about the goal space to learn the future state density, either through a representation (VIP methods) (Ghosh et al., 2018; Ma et al., 2022b) or through the value function (Choi et al., 2021). By observing this now-clarified relationship, we can not only compare the learned successor structures from GCRL to other methods that might more explicitly use successor measures like Forward Backward Representations (Touati et al., 2023) or PSM (Agarwal et al., 2025), but also utilize this to better understand the limitations of the optimal goal-reaching policy space and and uncompressed state, as compared to a parameterized space $\mathcal{Z}$, or a compressed space $\mathcal{X}$.

A.2 Mutual Information Skill Learning

Mutual Information Skill Learning (MISL) are a class of unsupervised RL algorithms that seeks to learn skill/option policies $\pi(a|s, z)$ that are conditioned on a latent variable $z \in Z$ representing the skills Zheng et al. (2024); Gregor et al. (2016); Park et al. (2023b); Campos et al. (2020); Laskin et al. (2022); Wang et al. (2024); Hu et al. (2024); Baumli et al. (2021). While previous MISL approaches often appear in different forms, they share a common objective of empowerment maximization, i.e. maximizing the mutual information $I(S; Z)$, where S represents some environment signal derived from the state visitation, such as the final state (s_T) Gregor et al. (2016), any state along a trajectory (s_t) Eysenbach et al. (2018b), or the transition (s_t, s_{t+1}) Baumli et al. (2021).

Direct optimization of this mutual information objective is intractable. Instead, it can be decomposed either in the reversed or forward form:

$$I(S; Z) = H(Z) - H(Z \mid S) \quad \text{// reverse} \tag{7}$$

$$= H(S) - H(S \mid Z) \quad \text{// forward} \tag{8}$$

which gives us different ways to approximate $I(S; Z)$ via variational inference. For example, DIAYN Eysenbach et al. (2018b) utilizes the reverse decomposition:

$$I(S; Z) = \mathbb{E}_{s,z \sim p(s,z)} [\log p(z \mid s)] - \mathbb{E}_{z \sim p(z)} [\log p(z)] \tag{9}$$

$$\geq \mathbb{E}_{s,z \sim p(s,z)} [\log q_\phi(z \mid s)] - \mathbb{E}_{z \sim p(z)} [\log p(z)] \tag{10}$$

resulting in the following intrinsic reward:

$$r_{\text{int}}(s, z) = \log q_\phi(z \mid s) \tag{11}$$

Some other algorithms resort instead to the forward decomposition Laskin et al. (2022); Campos et al. (2020), resulting in objectives that encourage both conditional state predictability $q_\phi(s \mid z)$ and the state diversity $H(S)$.

Recently, variations of the original mutual information objective have been proposed, including Wasserstein dependency measure Park et al. (2023c), factorized mutual information Hu et al. (2024), and conditional mutual information based on objects or interactions Wang et al. (2024).

Specifically, METRA Park et al. (2023c), introduces a metric-aware approach to unsupervised reinforcement learning. Instead of directly maximizing mutual information between skills and states, METRA employs the Wasserstein Dependency Measure (WDM) to capture the dependency between skills and states under a distance metric d. In METRA, the metric d is chosen to reflect the temporal distance between states, i.e., the minimum number of environment steps required to transition from one state to another. This choice of metric ensures that the learned skills are diverse in terms of their temporal dynamics, leading to behaviors that are not only distinguishable but also cover the state space effectively.

Under the lens of unification, mutual information skill learning methods represent the broad class of algorithms marrying exploration with successor measures. Through Theorem 4.6, we can view MISL methods as implicitly approximating the successor measure $M^{\pi_z}(s, a, s^+)$ by associating each skill z with a distinct mode in the future state distribution. Together, the skill-conditioned policies and the variational decoder represent a structured approximation of the underlying transition dynamics. This perspective reveals that MISL implicitly encodes the dynamics of the environment through its learned latent skills, and allows for comparison against explicit successor-measure-based methods like FB (Touati et al., 2023) or PSM (Silver et al., 2017).

A.3 Successor Features

Successor Features (Dayan, 1993; Barreto et al., 2017) are a class of multi-task RL algorithms that span rewards functions using state features as, $r = \phi w$ where ϕ are the state features and w is the task dependent linear weight. As a consequence,

$$\begin{aligned} Q^\pi(s, a) &= \mathbb{E}_\pi \left[\sum_t \gamma^t r(s_t) \right] \\ &= \mathbb{E}_\pi \left[\sum_t \gamma^t \phi(s_t) w \right] \\ &= \mathbb{E}_\pi \left[\sum_t \gamma^t \phi(s_t) \right] w \\ &= \psi^\pi(s, a) w \end{aligned} \tag{12}$$

where, $\psi^\pi(s, a)$ is called the successor feature and is defined as, $\psi^\pi(s, a) = \mathbb{E}_\pi \left[\sum_t \gamma^t \phi(s_t)\right]$.

Additionally, these methods align the latents of the optimal with the corresponding reward linear weights w i.e. $\pi_w = \arg\max \psi^{\pi_w}(s, a)w$. This linear dependence on the optimal policy reduces policy inference to simply finding the weight w corresponding to the reward function using linear regression, $w^* = \arg\min_w (\phi w - r)^2$.

A number of methods have been developed using this principle, starting from the ones using fixed, random or fourier features (Barreto et al., 2017; Zhu et al., 2024) to define the state features ϕ to others (Park et al., 2024; Agarwal et al., 2025) who have specialized objectives that add different inductive biases to these features.

A.4 Proto Successor Measures

Proto Successor Measures(PSM) (Agarwal et al., 2023) uses the observation that successor measures are obey linear Bellman Equations. As a result, they can be represented using an affine set. Successor Measures are hence represented as $M^\pi(s, a, s^+) = \sum_i \phi_i(s, a, s^+) w_i^\pi + b(s, a, s^+)$ where ϕ are the policy independent basis functions and b is the policy independent bias. w^π is a linear weight that depends on the policy. This parameterization enables an affine representation space containing the successor measures for all policies. Unlike successor features, PSM does not directly link the policy to its corresponding reward. Given any reward function, a simple constrained Linear Program needs to be solved to obtain w^*.

A.5 Proto Value Functions

Proto Value Functions refer to the class of spectral methods that linearize the value using the spectral decomposition of the graph Laplacian. They represent $V^\pi = \phi w^\pi$ where ϕ is independent of the policy while w^π is a policy-dependent linear weight. Mahadevan & Maggioni (2007) approximated the graph Laplacian using a random walk operator while some (Machado et al., 2017a; Farebrother et al., 2023) have used different objectives to directly approximate the eigenfunctions. Some of these works (Farebrother et al., 2023; Bellemare et al., 2019) simply minimize the regression loss against value functions of some auxiliary tasks.

A.6 Controllable Representations

A controllable representation is one in which only the features of the state that can change as a result of the policy are captured, and all other information is excluded. The controllable features are well described by the *endogenous* state of an Exogenous-MDP.

Definition A.1. (Exogenous Markov Decision Process (Efroni et al., 2022)). An exogenous-MDP (Exo-MDP) is a Block MDP where the observation s can be factored into two parts $s = (x, \xi)$ where $x \in \mathcal{X}$ is the endogenous state and $\xi \in \Xi$ is the exogenous state. The transitions of the exogenous and endogenous components of the state are independent as follows: $P(s'|s, a) = P(x'|x, a)P(\xi'|\xi)$.

Methods such as (Rudolph et al., 2024; Islam et al., 2023b;a; Efroni et al., 2022) attempt to learn an encoder $\phi : \mathcal{S} \to \mathcal{X}$ that only captures the endogenous components of the state. Notably, ACRO (Islam et al., 2023b) learns the encoder ϕ by performing a multi-step inverse dynamics prediction between two states k steps apart. The optimization is as follows,

$$\phi_\star \in \arg\max_{\phi\in\Phi} \mathbb{E}_{\substack{t\sim U(0,N)\\ k\sim U(0,K)}} \log\left(\mathbb{P}\left(a_t \mid \phi(s_t), \phi(s_{t+k})\right)\right), \tag{13}$$

where N is the maximum length of the episode and K is the time horizon of interest. A small modification to this objective, as shown in Levine et al. (2024) provably extracts the full N-step endogenous state. In contrast, Action-Bisimulation (Rudolph et al., 2024) learns a discounted infinite-horizon representation of controllability based on a minimal single-step inverse dynamics representation. The bisimulation metric (Ferns et al., 2011) is based on the bisimulation relation Givan et al. (2003) and learns a representation to approximately obey the following relation:

$$\begin{aligned} \psi(s_i) &= \psi(s_j) \\ P(\mathcal{G} \mid s_i, a) &= P(\mathcal{G} \mid s_j, a) \quad \forall a \in \mathcal{A}, \forall \mathcal{G} \in \mathcal{S}_{AB} \end{aligned} \tag{14}$$

where $\mathcal{S}_{AB}$ is the partition of $\mathcal{S}$ under the relation AB (the set of all groups $\mathcal{G}$ of equivalent states), and

$$P(\mathcal{G} \mid s, a) = \sum_{s'\in\mathcal{G}} p(s' \mid s, a),$$

and $\psi : \mathcal{S} \to \mathcal{Z}_{ss}$ is a representation such that $p(a \mid \psi(s), \psi(s')) = p(a \mid s, s')$ for all s, a, s'. The single-step representation ψ learns the features necessary for predicting the action taken to cause a transition. This representation is the basis of action-bisimulation because it filters out features that do not provide any signal to predict the action, i.e. anything that can be changed due to the agent's action.

While these controllable representation methods learn features that can be tied theoretically to the Unified Objective in Box 3.1, they do directly admit a policy. Instead, they provide efficient representations upon which downstream sequential decision-making tasks can be learned using RL.

B Proofs

B.1 Proof of Proposition 1

Proposition 3.1. *The algorithm presented in the Algorithm Box 3.1 is sufficient to produce optimal policies for any reward function.*

Proof. The algorithm contained in Algorithm Box 3.1 consists of two parts:

Pretraining: Learning $M^\pi(s, a, s^+),\ \forall\ s, a, s^+, \pi$.

Inference: Obtaining π^* for the given reward function using the pretrained representations.

The pretraining step simply ensures that M^π can be represented for any s, a, s^+, π.

As long as this is true, the question remains is if the inference step can produce optimal policies given that pretraining is true. To argue if the algorithm actually produces optimal policies for any reward function, we need to inspect inference.

The inference $Q^* = \max_\pi \sum_{s^+} M^\pi(s, a, s^+) r(s^+)$ produces a $Q^* \geq Q^\pi$ for all π. Hence for any reward function, the corresponding policy, $\max_\pi \sum_{s^+} M^\pi(s, a, s^+) r(s^+)$ produces the optimal policy as long as M^π correctly represents successor measures for all πs.

□

B.2 Proofs for Section 4.1

B.2.1 Proof of Theorem 3

Theorem 4.3. *With Π and $\mathcal{T}$ defined as per Assumptions 4.1 and 4.2, GCRL learns $Q^{\pi_z}(s, a) \propto M^{\pi_z}(s, a, z)$ for $s \in \mathcal{S}, z \in \mathcal{Z}, a \in \mathcal{A}$. The optimal policy inference for reward, r_z is π_z by construction.*

Proof. The proof follows simply from the definition of Q-function for goal conditioned RL. With reward function $r_z(s_t, a_t) = (1-\gamma)p(s_{t+1} = z|s_t, a_t)$, the Q-function is defined as:

$$Q^{\pi_z}(s, a) = (1-\gamma)\mathbb{E}_{\pi_z}\left[\sum_{t=0}^{\infty}[\gamma^t p(s_{t+1} = z|s_t, a_t)]\right] \tag{15}$$

$$= M^{\pi_z}(s, a, z) \tag{16}$$

□

B.3 Proofs for Section 4.2

B.3.1 Proof of Theorem 6

Theorem 4.6. *For Π defined using Assumption 4.4 and $\mathcal{T}$ defined using Assumption 4.5, MISL objectives learn $M^{\pi_z}(s, s^+) = \frac{q(z|s^+,s)p(s^+|s)}{p(z)}$ for $s \in \mu$, $a \sim \pi_z(\cdot|s \sim \mu)$ and $s^+ \in \mathcal{S}$. The policy inference can be performed by searching through the space of $z \in \mathcal{Z}$ for rewards defined in $\mathcal{T}$.*

Proof. Start with the MISL conditional distribution $p(z|s^+, s)$, where s is the starting state and typically omitted from MISL formulations, and s^+ is the current state, which is approximated by the variational distribution $q(z|s^+, s)$. Applying bayes rule gives:

$$p(z|s^+, s)p(s^+|s) = p(s^+|z, s)p(z|s) \tag{17}$$

$$\frac{p(z|s^+, s)p(s^+|s)}{p(z|s)} = p(s^+|z, s) \tag{18}$$

$$\frac{q(z|s^+, s)p(s^+|s)}{p(z)} \approx p(s^+|z, s) \tag{19}$$

$$\mathbb{E}_{\pi_z}\left[\frac{q(z|s^+, s)p(s^+|s)}{p(z)}\right] \approx M^{\pi_z}(s, s^+) \tag{20}$$

The second line replaces $p(z|s)$ with $p(z)$, because the skills in MISL are sampled independently of the starting state. $p(s^+|z, s)$ is the probability of seeing a future state s^+ starting from state s and following a skill z. $p(s^+|z, s) = (1-\gamma)\sum_{t>0} p(s_t = s^+|s, z) = M^{\pi_z}(s, s^+)$. The final transformation utilizes the fact that z is the parameterization of a policy. □

Remark. While $\frac{q(z|s^+,s)p(s^+|s)}{p(z)}$ appears to be quite messy, note that the state covering nature of MISL which arises from policies optimizing the reward $r(s^+) = \log q(z|s^+, s) + \log p(z)$ actually helps to remove the complexity. In particular, if the skills are successfully state covering from starting state s, then $p(s^+|s) = p(z)$, that is the likelihood of reaching a state s^+ from state s will match the likelihood of the corresponding skill being sampled, which is just $p(z)$. This leaves: $q(z|s^+, s) \approx M^{\pi_z}(s, s^+)$, where q is a variational approximation of the future state density.

B.3.2 Proof of Theorem 7

Theorem 4.8. *(Choi et al., 2021) For $\mathcal{Z} = \mathcal{S}$, GCRL with $r(s|z) = -\frac{1}{\sigma^2}\|z - s\|$ is the same as solving the MISL objective with the variational distribution, $q(z|s) = \mathcal{N}(z - s, \sigma^2)$.*

Proof. This proof can be found in Choi et al. (2021) and is summarized here. Notice that the reward for MISL policy learning is $\log q(z|s^+, s) - \log p(z)$. Assigning the space of z to equal s, $\mathcal{Z} = \mathcal{S}$, we can then replace $q(z|s^+, s) = \log \exp(-\frac{\|z-s^+\|}{\sigma^2}) - \log(2\pi)$. Replace this value back into the reward function for GCRL, and this gives $q(z|s^+, s) = \log \exp(-\frac{\|z-s^+\|}{\sigma^2}) - \log(2\pi) + \log(2\pi) = -\frac{\|z-s^+\|}{\sigma^2}$, when $p(z)$ is a unit normal distribution. This completes the proof. □

B.3.3 Proof of Theorem 8

Theorem 4.7. *(Zheng et al., 2025) Given a critic function, $f : \mathcal{S} \times \mathcal{S} \times \mathcal{Z} \to \mathbb{R}$, $I^\pi(S, S'; Z) \geq \mathbb{E}_{p^\pi(s,s',z)}[f(s, s', z)] - \mathbb{E}_{p^\pi(s,s')}[\log \mathbb{E}_{p(z)}[e^{f(s,s',z)}]]$ where the right hand side is the variational lower bound: $(VLB(f, \pi))$*

Proof. This proof is adapted from Zheng et al. (2025). Starting from the standard information lower bound adapted for (S, S^+) and Z.

$$I^\pi(S, S^+; Z) \geq \mathbb{E}_\pi[\log q(z|s, s^+)] + H(Z) \tag{21}$$

$$\geq \mathbb{E}_{s,s^+\sim\rho(\pi), z\sim p(z)}[f(s, s^+, z)] - \mathbb{E}_{s,s^+\sim\pi}[\log \mathbb{E}_{z\sim p(z)}[\exp(f(s, s^+, z))]] \tag{22}$$

The first equation is the Barber-Agakov Inequality Barber & Agakov (2004) applied to our setting. The second plugs in an energy based variational family, where $q(z|s, s^+) = \frac{p(x)\exp(f(s,s^+,z))}{\mathbb{E}_{p(z)}[f(s,s^+,z)]}$ according to Poole et al. (2019a). Thus, the information objective of MISL is lower bounded by a successor representation on s, s^+ and z. □

B.3.4 Additional Equivalences

Theorem B.1. *Parameterizing $f(s, s', z)$ in Theorem 4.7 as $f(s, s', z) = (\phi(s) - \phi(s'))^T z$, METRA Park et al. (2023c) is obtained as an approximation to $VLB(\phi, \pi)$.*

Proof. This proof is adapted from Zheng et al. (2025). Starting from the previous observation and replacing s^+ with s' gives:

$$I^\pi(S, S^+; Z) \geq \mathbb{E}_\pi[f(s, s^+, z)] - \mathbb{E}_{s,s^+\sim\pi}[\log \mathbb{E}_{z\sim p(z)}[\exp(f(s, s^+, z))]] \tag{23}$$

$$\geq \mathbb{E}_\pi[(\phi(s) - \phi(s'))^\top z] - \mathbb{E}_{s,s^+\sim\pi}[\log \mathbb{E}_{z\sim p(z)}[\exp(\phi(s) - \phi(s'))^\top z]] \tag{24}$$

$$\approx \min_{\lambda\geq 0} \mathbb{E}_\pi[(\phi(s) - \phi(s'))^\top z] - \lambda(d)(1 - \mathbb{E}_{s,s'\sim\rho(\pi)}[\|\phi(s) - \phi(s')\|^2] \tag{25}$$

Where the final line replaces the log-sum-exponential term with a second order taylor approximation.

□

B.4 Proofs for Section 4.3

B.4.1 Proof of Theorem 11

Theorem 4.11. *With Π and $\mathcal{T}$ as defined by Assumptions 4.10 and 4.9, SF methods learn $M^{\pi_z}(s, a, s^+) = \psi(s, a, z)(\Phi^\top\Phi)^{-1}\Phi^\top, \forall\ s, s^+ \in \mathcal{S}$ and $a \in \mathcal{A}$. The inference on any reward function in $\mathcal{T}$ requires solving a linear regression problem, $z^* = \arg\min_z (r - \Phi^\top z)^2$.*

Proof. Successor Features assume $r = \phi z$ for some linear weight z. This assumption directly leads to $Q^\pi(s,a) = \psi^\pi(s,a)z$ where ψ^π is the successor feature using the state features ϕ (See Section A.3).

As $r = \phi z, \implies z = (\phi^T\phi)^{-1}\phi^T r$.

Substituting in Q^{π_z} (following from Section A.3, π is conditioned on z),

$$\begin{aligned} &Q^{\pi_z}(s,a) = \psi(s,a,z)z \\ \implies &Q^{\pi_z}(s,a) = \psi(s,a,z)(\phi^T\phi)^{-1}\phi^T r \end{aligned} \tag{26}$$

Following from $Q^{\pi_z} = M^{\pi_z} r$ for all r, it can be shown that $M^\pi = \psi(s,a,z)(\phi^T\phi)^{-1}\phi^T$. □

B.4.2 Proof of Theorem 12

Theorem 4.12. *If the successor measure is parameterized as, $M^\pi(s,a,s+) = F(s,a,z)^\top B(s^+)$, with $B(s^+) = (\Phi^\top\Phi)^{-1}\phi^\top(s^+)$ and $F(s,a,z) = \psi(s,a,z)$, the algorithm in Theorem 4.11 reduces to the FB algorithm (Touati & Ollivier, 2021). The policy inference simply becomes $z^* = Br$.*

Proof. Forward Backward representations (Touati & Ollivier, 2021) represents $M^{\pi_z}(s,a,s^+) = F(s,a,z)^\top B(s^+)$.

As a result, $Q^{\pi_z}(s,a) = \sum_{s^+} M^{\pi_z}(s,a,s^+) r_z(s^+) = \sum_{s^+} F(s,a,z)^\top B(s^+) r(s^+)$.

(Touati et al., 2023) has shown that $F(s,a,z)$ is the successor feature for the state feature $(B^\top B)^{-1}B^\top$. It can be similarly shown that, the backward network in FB is the same as $(\phi^T\phi)^{-1}\phi^T$ in the SF parameterization of M^π. □

B.4.3 Proof of Theorem 13

Theorem 4.13. *If $\phi = \arg\min_\phi \mathbb{E}_{s,s',g}[\ell_\tau(||\phi(s) - \phi(g)|| - \mathbb{1}_{s\neq g} - \gamma||\phi(s') - \phi(g)||)]$ in Theorem 4.11, with $r(s,s',z) = (\phi(s) - \phi(s'))^\top z$, the resulting algorithm is HILP (Park et al., 2024).*

Proof. The HILP algorithm (Park et al., 2024) consists of three major steps: (1) Learning a state representation ϕ, (2) Defining reward functions using ϕ and a linear weight z and (3) Training π_z to maximize r_z.

The first step of learning a state representation uses the following optimization,

$$\phi^* = \arg\min_\phi \mathbb{E}_{s,s',g}[\ell_\tau(||\phi(s) - \phi(g)|| - \mathbb{1}_{s\neq g} - \gamma||\phi(s') - \phi(g)||)] \tag{27}$$

The second step, defines a reward function $r(s,s',z) = \phi(s,s')z = (\phi(s) - \phi(s)')z$.

Finally, the final step requires training π_z for corresponding r_z. This is achieved in practice by parameterizing the Q-function using successor features.

Hence, HILP algorithm is an SF based method with state features, ϕ, trained using Equation 27. □

B.4.4 Proof of Theorem 14

Theorem 4.14. *If $\phi = \arg\max_\phi \mathbb{E}_{p^\pi(s,s',z)}[(\phi(s) - \phi(s'))^\top z] - \mathbb{E}_{p^\pi(s,s')}[\log \mathbb{E}_{p(z)}[e^{(\phi(s)-\phi(s'))^\top z}]]$, in Theorem 4.11, with $r(s,s',z) = (\phi(s) - \phi(s'))^\top z$, the resulting algorithm is CSF (Zheng et al., 2025).*

Proof. Similar to the previous proof, CSF(Zheng et al., 2025) introduces a SF based algorithm that uses a MISL inspired objective to train state features, ϕ,

$$\phi = \arg\max_\phi \mathbb{E}_{p^\pi(s,s',z)}[(\phi(s) - \phi(s'))^\top z] - \mathbb{E}_{p^\pi(s,s')}[\log \mathbb{E}_{p(z)}[e^{(\phi(s)-\phi(s'))^\top z}]] \tag{28}$$

Like HILP, CSF defines its reward function for SF as a linear span of the basis, $r(s, s', z) = \phi(s, s')z = (\phi(s) - \phi(s)')z$. □

B.5 Proofs for Section 4.4

B.5.1 Proof of Theorem 16

Theorem 4.16. *PSM learns* $M^{\pi_z}(s, a, s^+) = \sum_i \phi_i(s, a, s^+) w_i^{\pi_z} + b(s, a, s^+)$ *for* $\pi_z \in \Pi$ *as defined in Assumption 4.15.*

Proof. Proto Successor Measures (PSM) (Agarwal et al., 2025) parametrizes successor measures using an affine decomposition i.e. using basis and bias functions. Theorem 16 is a direct consequence of the parameterization. □

B.5.2 Proof of Theorem 17

Theorem 4.17. *For the PSM representation* $M^\pi(s, a, s^+) = \phi(s, a, s^+)w^\pi + b(s, a, s^+)$ *and* $\phi(s, a, s^+) = \phi_\psi(s, a)^T \varphi(s^+)$*, the successor feature* $\psi^\pi(s, a) = \phi_\psi(s, a)w^\pi$ *for the state feature* $\varphi(s)^T(\mathbb{E}_\rho(\varphi\varphi^T))^{-1}$.

Proof. The proof for this theorem is adapted from Agarwal et al. (2025).

According to the PSM parameterization, $M^\pi(s, a, s^+)$ can be represented as $\phi(s, a, s^+)w^\pi$ (dropping the bias term for simplicity. It can be thought of as absorbing the bias term into the basis. If $\phi(s, a, s^+) = \phi_\psi(s, a)^T \phi_s(s^+)$, for some ϕ_ψ and ϕ_s,

$$
\begin{aligned}
M^\pi(s, a, s^+) &= \sum_i \sum_j \phi_\psi(s, a)_{ij} \phi_s(s^+)_j w_i^\pi \\
\implies M^\pi(s, a, s^+) &= \sum_j \sum_i \phi_\psi(s, a)_{ij} w_i^\pi \phi_s(s^+)_j \\
\implies M^\pi(s, a, s^+) &= \sum_j \phi_\psi(s, a)_j^T w^\pi \phi_s(s^+)_j \\
\implies M^\pi(s, a, s^+) &= \sum_j \psi^\pi(s, a)_j \phi_s(s^+)_j \qquad \text{(Writing } \phi_\psi(s, a)^T w^\pi \text{ as } \psi^\pi(s, a)) \\
\implies M^\pi(s, a, s^+) &= \psi^\pi(s, a)^T \phi_s(s^+)
\end{aligned}
$$

From Theorem 4.12, $\psi^\pi(s, a)$ is the successor feature for the basic feature $\phi_s(s)^T(\phi_s\phi_s^T)^{-1}$.

□

B.6 Proofs for Section 4.5

B.6.1 Proof of Theorem 20

Theorem 4.20. *The eigenvectors used by PVFs are the same as that of* $M^{\pi_U}(s, s^+)$*. Therefore, PVFs learn* $M^{\pi_U}(s, s^+) = \phi w$*. The policy inference for a reward function in the class* $\mathcal{T}$ *follows from the LSPI algorithm.*

Proof. PVFs learn eigenvectors for the graph laplacian given by,

$$\mathcal{L} = D - A \tag{29}$$

where D is the degree matrix and A is the adjacency matrix.

The normalized graph laplacian is given by, $I - D^{-1/2}AD^{1/2}$. The random walk operator is given by,

$$L = I - T \tag{30}$$

where $T = D^{-1}A$

The Successor Representation(SR) (Ψ^π) is a quantity related to successor measures as,

$$\Psi^\pi(s, s') = \sum_{t>0} \gamma^t \mathbb{P}(s_t = s' | s_0 = s, \pi) \tag{31}$$

Clearly, $M^\pi(s, s^+)$ is the same as $\Psi^\pi(s, s^+)$. Additionally, for a value function, $V^\pi = \Psi^\pi r = (I - \gamma P^\pi)^{-1} r$. This implies, $\Psi^\pi = (I - \gamma P^\pi)^{-1}$.

The eigen-decomposition of SR and the graph laplacians have been extensively studies by Machado et al. (2017b); Stachenfeld et al. (2014); Farebrother et al. (2023). They have shown that if ϕ is an eigenvector of the random walk operator (L), $\gamma\phi$ is the corresponding eigenvector for discounted random walk laplacian, $I - \gamma T$. And $(I - \gamma T)^{-1}$ has the corresponding eigenvector of $\gamma D^{-1/2}\phi$.

Hence, *if π is uniform, i.e. $P^\pi = T$*, PVFs which finds the eigenvectors for the graph laplacians (random walk or normalized), also correspondingly obtain the eigenvectors for $M^{\pi_U}(s, s^+)$.

□

B.6.2 Comparison with PSM

PSM (Agarwal et al., 2025) has introduced the following theorem that compares the representative powers of PVFs compared to PSM:

Theorem B.2. *(Agarwal et al., 2025) Given a d-dimensional basis $\mathbf{B} : \mathbb{R}^n \to \mathbb{R}^d$, define $span\{\mathbf{B}\}$ as the span of all linear combinations of basis $\mathbf{B}$. Further define $span\{\mathbf{B}r\}$ as the span of inner products of all linear combinations of basis $\mathbf{B}$ and all possible reward functions r. Let $span\{\Phi^{vf}\}$ denote the space of the value functions spanned by Φ^{vf} while $\{span\{\Phi\}r\}$ denotes the space of value functions using the successor measures spanned by Φ. For the same dimensionality of task (policy or reward) independent basis, $span\{\Phi^{vf}\} \subseteq \{span\{\Phi\}r\}$ for some Φ.*

The theorem suggests that given the same number of dimensions, d, any method that spans the space of successor measures represents a larger set of value functions from the methods that span the space of value functions. We present a short adaptation of the proof from Agarwal et al. (2025).

Proof. We need to show that any element that belongs to the set $span\{\Phi^{vf}\}$ also belongs to the set $\{span\{\Phi\}r\}$.

Any element belonging to the set $\{span\{\Phi^{vf}\}\}$ is represented by,

$$V^\pi(s) = \sum_i \beta_i^\pi \Phi_i^{vf}(s).$$

Similarly, any element in $\{span\{\Phi\}r\}$ can be represented by,

$$V^\pi(s) = \sum_i w_i^\pi \sum_{s'} \Phi_i(s, s') r(s')$$

It is possible to show that for every element in $\{span\{\Phi^{vf}\}\}$, there exists some element in $\{span\{\Phi\}r\}$ but the reverse is not true. Only when $\Phi_i(s, s') = \sigma_i(s)\eta_i(s')$ for some σ and η, can an element from $\{span\{\Phi\}r\}$ is present in $\{span\{\Phi^{vf}\}\}$.

□

B.7 Proofs for Section 4.6

B.7.1 Proof of Theorem 24

Theorem 4.24. *Multi-step inverse methods like Lamb et al. (2022); Islam et al. (2023a); Levine et al. (2024), model* $M_K^{\pi_\beta}, \ \forall\, s \in \mathcal{S}, \ a \in \mathcal{A}, \ s^+ \in \mathcal{S}$ *as* $M_K^{\pi_\beta}(s, a, s^+) = \frac{f(a|s,s^+)p^{\pi_\beta}(s^+|s)}{\pi_\beta(a|s)}$.

Proof. Starting from the definition of K step inverse dynamics $p(a|s, s^+)$, where s^+ is a state K steps distant, π_β is the behavior policy and $f(a, s, s^+)$ is the learned inverse dynamics, and the definition of $M_K^{\pi_\beta}(s, a, s^+) = \mathbb{E}_{\pi_\beta} p(s^{t+k} = s^+|s^t, a^t)$, we can apply bayes rule to achieve the transformations:

$$p(a|s, s^+, \pi_\beta)p(s^+|s, \pi_\beta) = p(s^+|a, s, \pi_\beta)p(a|s, \pi_\beta) \tag{32}$$

$$\frac{p(a|s, s^+, \pi_\beta)p(s^+|s, \pi_\beta)}{p(a|s, \pi_\beta)} = p(s^+|a, s, \pi_\beta) \tag{33}$$

$$\frac{p(a|s, s^+, \pi_\beta)p(s^+|s, \pi_\beta)}{\pi_\beta(a|s, \pi_\beta)} = p(s^+|a, s, \pi_\beta) \tag{34}$$

$$\frac{f(a, s, s^+)p(s^+|s, \pi_\beta)}{\pi_\beta(a|s)} \approx p(s^+|a, s) \tag{35}$$

$$\frac{f(a|s, s^+)p(s^+|s, \pi_\beta)}{\pi_\beta(a|s)} \approx M_K^{\pi_\beta}(s, a, s^+) \tag{36}$$

Notice that line 3 utilizes the fact that $p(a|s)$ in the offline distribution is the definition of the behavior policy, and line 4 uses the learned inverse dynamics to approximate the true inverse probability, where the learned inverse dynamics are learned according to π_β. □

B.7.2 Proof of Theorem 25

Theorem 4.25. *In Action-Bisimulation (Rudolph et al., 2024),* $||\phi(s_1) - \phi(s_2)|| = 0 \Leftrightarrow M^{\pi_U}(s_1, a, s^+) = M^{\pi_U}(s_2, a, s^+), \forall\, a \in \mathcal{A}, \ s^+ \in \mathcal{S}$ *where* π_U *is a uniformly random policy.*

Proof. Consider the bisimulation equality for action bisimulation, where $\rho(\pi_U, s)$ is the distribution of trajectories following the uniform policy from state s:

$$\|\phi(s_1) - \phi(s_2)\| = \|\varphi(s_1) - \varphi(s_2)\| + \gamma \mathbb{E}_{\pi_u}\left[\mathcal{W}(f(\cdot|s_1, a), f(\cdot|s_2, a))\right] \tag{37}$$

$$\|\phi(s_1) - \phi(s_2)\| = \mathbb{E}_{\tau_1 \sim \rho(\pi_U, s_1), \tau_2 \sim \rho(\pi_U, s_2)}\left[\sum_{t=0}^{\infty} \gamma^t \|\varphi(s_1^t) - \varphi(s_2^t)\|^2\right] \tag{38}$$

The conversion between lines 1-2 simply unrolls the boostrapped wasserstein term (recall that $f : \mathcal{S} \times \mathcal{A} \to \Delta(\phi(\mathcal{S}))$, or a distribution over $\phi(s')$. Notice that the last term implies that $\|\phi(s_1) - \phi(s_2)\| = 0$ only if sum of all possible future values of $\|\varphi(s_1^t) - \varphi(s_2^t)\| = 0$, for all possible sequences of states. If this is true, since $\varphi(s)$ captures all the myopic action-relevant (and thus dynamic variability) information, $M^{\pi_U}(s_1, a, s^+) = M^{\pi_U}(s_2, a, s^+)$ for all future trajectories.

In the case where $M^{\pi_U}(s_1, a, s^+) = M^{\pi_U}(s_2, a, s^+)$, this implies also that all future distributions are the same, which means that the future trajectories match, or in other words that there is a one-to-one equivalence between $\rho(\pi_U, s_1) \equiv \rho(\pi_U, s_2) \equiv \rho(\pi_U, s_{1/2})$. Then:

$$M^{\pi_U}(s_1, a, s^+) - M^{\pi_U}(s_2, a, s^+) = 0 \quad \Rightarrow \tag{39}$$

$$E_{\tau_1 \sim \rho(\pi_U, s_1), \tau_2 \sim \rho(\pi_U, s_2)} \left[\sum_{t=0}^{\infty} \gamma^t \|\varphi(s_1^t) - \varphi(s_2^t)\|^2 \right] =$$

$$E_{\tau_{1/2} \sim \rho(\pi_U, s_{1/2})} \left[\sum_{t=0}^{\infty} \gamma^t \|\varphi(s_{1/2}^t) - \varphi(s_{1/2}^t)\|^2 \right] \quad \Rightarrow \tag{40}$$

$$\|\phi(s_1) - \phi(s_2)\| = 0 \tag{41}$$

Because the trajectories from s_1 and s_2 can be sampled equivalently. Since both $\|\phi(s_1) - \phi(s_2)\| = 0 \Rightarrow M^{\pi_U}(s_1, a, s^+) - M^{\pi_U}(s_2, a, s^+) = 0$ and $M^{\pi_U}(s_1, a, s^+) - M^{\pi_U}(s_2, a, s^+) = 0 \Rightarrow \|\phi(s_1) - \phi(s_2)\| = 0$, this means $\|\phi(s_1) - \phi(s_2)\| = 0 \iff M^{\pi_U}(s_1, a, s^+) - M^{\pi_U}(s_2, a, s^+) = 0$ □

B.8 State Equivalences

In Section 5, we introduced the notion that every method explicitly or through some approximations, produces state abstractions where the state space is compressed based on state equivalences. We re-introduce state-equivalences in the practical settings:

We want to learn $\phi : \mathcal{S} \to \mathcal{X}$ such that, $M^\pi(s, a, s^+) = M^\pi(\phi(s), a, \phi(s^+))$. Additionally, $\phi(s_1) = \phi(s_2)$ iff $M^\pi(\phi(s_1), a, \phi(s^+)) = M^\pi(\phi(s_2), a, \phi(s^+))$.

We mentioned that all these methods compress states based on the “distance" between the abstractions $d(\phi(s_1), \phi(s_2))$ as being proportional to $p(s_1 = s_2)$. We shall discuss the “distance" used by each of these URL algorithms:

Goal Conditioned RL: Goal Conditioned Value Functions have often been shown to be quasimetrics (Wang et al., 2023) in special cases. But, in most general settings, goal conditioned value functions follow the triangle inequality (Liu et al., 2023). As a result, a number of methods (Ma et al., 2022b; Park et al., 2024) have represented value functions using L2 distances: $V(s, g) = -||\phi(s) - \phi(g)||$. These define the distances in GCRL space.

Mutual Information Skill Learning: MISL works compress the state representations using skills. Two states are similar if they impose the same skills. Hence the two distributions, $q(z|s_1)$ and $q(z|s_2)$ are the same if the states are equivalent (from a MISL perspective). Which means $D_{KL}(q(z|s_1)||q(z|s_2))$ represents the distance between the skill distributions for the two states s_1 and s_2.

Successor Features: SFs (and approximated PSM) also produce state abstractions in the form of state features. Successor measures are defined as, $M^\pi(s, a, s^+) = \sum_{t>0} p^\pi(s_t = s^+|s_0 = s, a_0 = a) = \mathbb{E}_\pi[\sum_{t>0} p(s_t = s^+|s_0 = s, a_0 = a)]$. Successor Features alternately define $M^\pi = \mathbb{E}_\pi[\sum_{t>0} \phi(s_t)^\top \phi(s^+)]$. Both these are equivalent for all π. This implies the state equivalences, $p(s_1 = s_2)$ is given by $\phi(s_1)^\top \phi(s_2)$ in case of SFs. This explains why methods (Touati et al., 2023; Touati & Ollivier, 2021) often impose orthonormality in some form in ϕ.

Proto Value Functions: PVFs represent a basis for the value functions. Any two states being the same would induce the same components of the basis. Which means $\phi(s) \in \mathbb{R}^d$ will be parallel. Hence, similar to SFs, PVFs also use cosine distance, $\phi(s_1)^\top \phi(s_2)$.

Controllable Representations: While Islam et al. (2023b); Lamb et al. (2022); Levine et al. (2024) directly optimize for state compression using the definition (by implicitly using successor measures), methods like Rudolph et al. (2024) use an L2 distance to characterize distance between two states as discussed in Theorem 4.25.

C Additional Unsupervised RL Methods

While this work draws equivalences between several major classes of Unsupervised RL algorithms, we certainly do not cover all possible methods. This is not because we do not believe that these methods have relevant equivalences, but rather for time and space constraints. In this section we mention a number of additional directions that we believe share links, if not explicit reductions, to the successor measure and state equivalence abstraction. In representation learning, Bootstrap your own latent Grill et al. (2020) and Contrastive RL Eysenbach et al. (2022b) show close similarities with both action representations and successor features. Empowerment Klyubin et al. (2005); Eysenbach et al. (2018b) has long been linked to mutual information skills, while the graph Laplacian Machado et al. (2017a) and reward-free world models Ha & Schmidhuber (2018); Fujimoto et al. (2025) show close ties to spectral methods. Inverse reinforcement learning Ng et al. (2000); Ghasemipour et al. (2020) and even behavior cloning Ke et al. (2021); Brohan et al. (2023) might be seen as identifying a particular expert visitation distribution. Finally, exploration methods utilize estimates of the current state visitation distribution either through counts Bellemare et al. (2016) or curiosity Pathak et al. (2017), and have close ties with mutual information objectives. As we can see, this work just begins a process of finding similarities and differences between existing reward-free methods. Through this work, we hope to clarify the avenues for cross-pollination and improvement in identifying the best tools when learning policies in complex environments.