Abstract and Explore: A Novel Behavioral Metric with Cyclic Dynamics in Reinforcement Learning

Anjie Zhu[1], Peng-Fei Zhang[2], Ruihong Qiu[2], Zetao Zheng[1], Zi Huang[2], Jie Shao[1]*

[1]University of Electronic Science and Technology of China, China
[2]The University of Queensland, Australia
{anjiezhu,ztzheng}@std.uestc.edu.cn, mima.zpf@gmail.com, r.qiu@uq.edu.au, huang@itee.uq.edu.au, shaojie@uestc.edu.cn

Abstract

Intrinsic motivation lies at the heart of the exploration of reinforcement learning, which is primarily driven by the agent's inherent satisfaction rather than external feedback from the environment. However, in recent more challenging procedurally-generated environments with high stochasticity and uninformative extrinsic rewards, we identify two significant issues of applying intrinsic motivation. (1) *State representation collapse*: In existing methods, the learned representations within intrinsic motivation have a high probability to neglect the distinction among different states and be distracted by the task-irrelevant information brought by the stochasticity. (2) *Insufficient interrelation among dynamics*: Unsuccessful guidance provided by the uninformative extrinsic reward makes the dynamics learning in intrinsic motivation less effective. In light of the above observations, a novel *B*ehavioral metric with *C*yclic *D*ynamics (BCD) is proposed, which considers both cumulative and immediate effects and facilitates the abstraction and exploration of the agent. For the behavioral metric, the successor feature is utilized to reveal the expected future rewards and alleviate the heavy reliance of previous methods on extrinsic rewards. Moreover, the latent variable and vector quantization techniques are employed to enable an accurate measurement of the transition function in a discrete and interpretable manner. In addition, cyclic dynamics is established to capture the interrelations between state and action, thereby providing a thorough awareness of environmental dynamics. Extensive experiments conducted on procedurally-generated environments demonstrate the state-of-the-art performance of our proposed BCD.

Introduction

Reinforcement learning (RL) has emerged as a powerful framework for training intelligent agents to make optimal decisions in a wide range of real-world scenarios (Levine et al. 2016; Kendall et al. 2019; Schrittwieser et al. 2020; Afsar, Crump, and Far 2023). Within the realm of RL, exploration plays a vital role in encouraging the agent to uncover and comprehend the environment with new experiences to maximize long-term rewards, in addition to solely exploiting the current mastered knowledge. Researchers have proposed various exploration strategies, ranging from classical methods such as ϵ-greedy and upper confidence bounds (UCB) (Lai and Robbins 1985) to more recent advancements such as novelty (Burda et al. 2019; Zhang et al. 2021b) and uncertainty-oriented exploration (Osband et al. 2019; Moerland, Broekens, and Jonker 2017).

Among the recent methods, the dominant strategy is to leverage the intrinsic motivation, which draws inspiration from the innate inclination of humans when making decisions. The agent is motivated to seek out unfamiliar regions or discover new potentially rewarding experiences by employing techniques. For example, through estimating the prediction errors of the environmental dynamics (Pathak et al. 2017), such incorrectly predicted regions are regarded as unfamiliar ones. By estimating the count of the state visitation (Zhang et al. 2021b), those with a lower count are qualified for the novel ones. Intrinsic motivation furnishes qualitative guidance to the agent in their pursuit of exploring the environment autonomously and learning effectively.

While the intrinsic motivation has attracted significant attention from the research community, it becomes notably intricate when being applied to procedurally-generated environments with stochasticity and uninformative rewards. In this situation, two notable issues can be identified when training an RL agent. (1) *State representation collapse*. Due to the inherent stochasticity of the environment where every episode is randomly constructed, the agent would easily be confused with environmental variations such as different maze layouts and varying positions of key elements. Unfortunately, previous methods may inadvertently capture and emphasize on task-irrelevant (e.g., background objects) or noisy information. In consequence, the agent is hard to correctly discriminate the state representations and may perceive functionally distinct states as similar, leading to suboptimal decisions. (2) *Insufficient interrelation among dynamics*. The dynamics refers to the underlying transitions between states and actions within the environment. Existing methods often overlook the dependencies among the dynamics, which only focus on forward or inverse dynamics learning (Pathak et al. 2017). Besides stochasticity, unsuccessful guidance stemming from almost zero or constant extrinsic rewards also makes it more difficult to accurately model and understand the complex relationship between states and actions. Without an accurate perception of the environmental dynamics, the exploration and decision-making of the agent may be ineffective.

*Corresponding author: Jie Shao.

To address these issues, this paper aims to simultaneously optimize representation and dynamics learning, ultimately heading to effective exploration and optimal decision-making strategies. Intuitively, aiming at the issue of *state representation collapse*, it is expected that the representation is invariant to irrelevant information and behaviorally similar states can be grouped together. Moreover, integrating the temporal abstraction brings the agent a complementary of both uninformative extrinsic reward and long-term impact. Then, a robust and informative state representation is provided, alleviating the collapse issue to some extent. For the issue of *insufficient interrelation among dynamics*, there are more complex relationships in procedurally-generated environments worthwhile to dive into. For instance, backward dynamics learning which predicts the representation of the state based on the action and representation of the next state emerges as a potential tool. Thus, a more holistic view of the environmental dynamics is developed to mitigate the insufficiency issue, simultaneously assisting in representation learning. With the basis of effective representation and dynamics learning, optimal exploration and decision-making will be accomplished.

In light of the above discussions, in this paper we present a novel *B*ehavioral metric with *C*yclic *D*ynamics (BCD), which facilities the effective abstraction and exploration. Specifically, in the behavioral metric, the distance between two state representations corresponds to the discrepancy in their behavior which is jointly determined by the difference in estimated rewards and transition functions. For the reward part, we estimate it with the successor feature which serves as a temporal abstraction and involves the expected future cumulative rewards associated with each state. For the transition dynamics part, we leverage the latent variable and vector quantization technique to obtain the discrete representation, allowing us to calculate the distance between the clusters of each state and measure their behavioral similarity accurately. Our behavioral metric presents an effective way to achieve state abstraction. Moreover, regarding insufficient interrelation among dynamics, the backward dynamics is incorporated with forward and inverse dynamics to establish the cyclic transition dynamics.

Overall, our contributions are summarized as follows:

- The state representation collapse and insufficient interrelation among dynamics issues are identified and analyzed in procedurally-generated environments.
- A novel behavioral metric is built to comprehensively account for both immediate and cumulative effects of decisions in a discrete and interpretable manner.
- Cyclic dynamics is developed to assist the agent to gain a more holistic understanding of the environmental dynamics.
- Extensive experiments demonstrate the superiority of our proposed BCD in complex and hard environments.

Related Work

Intrinsic Motivation. There are two main lines of intrinsic motivation in exploration, novelty-based methods and prediction-based methods. Novelty-based methods focus on encouraging agents to visit the states that they have not or rarely visited before. Random network distillation (RND) (Burda et al. 2019) introduces the state novelty with a representation network to predict another representation network with the fixed random initialization. Never give up (NGU) (Badia et al. 2020) unifies the intra-episode and inter-episode novelty, which promotes exploration within episodes by encouraging the agent to visit diverse states and provides a global measurement across the learning process respectively. Rewarding impact-driven exploration (RIDE) (Raileanu and Rocktäschel 2020) proposes to encourage the large change in representation in a latent space. Adversarially guided actor-critic (AGAC) (Flet-Berliac et al. 2021) maximizes the discrepancy between its action log-probabilities and those predicted by the adversary network. NovelD (Zhang et al. 2021b) only rewards the agent for its first visit to the novel state at the episode level. Regulated difference of inverse visitation counts of consecutive states in a trajectory is leveraged in Bebold (Zhang et al. 2020) where visitation count is approximated by RND.

Another line of research resorts to learning the dynamics model and constructing the intrinsic reward through prediction error. If the prediction of a model is inaccurate for a specific state, it may suggest that the given state has been encountered infrequently, resulting in a high intrinsic reward signal. Intrinsinc curiosity module (ICM) (Pathak et al. 2017) leverages the prediction error between the representation of the next state and the predicted one. Variational dynamic model (VDM) (Bai et al. 2023) introduces a variational objective where latent variables encode the multimodality and stochasticity of the underlying dynamics. However, previous methods are susceptible to the influence of representation accuracy in stochastic environments.

Representation Learning in RL. Self-supervised representation learning has emerged as a prominent technique in reinforcement learning, playing a crucial role in enhancing the effectiveness and efficiency of agents. Many efforts have been made to incorporate auxiliary tasks to learn representation, e.g., Yarats et al. (2021); Hafner et al. (2019) which aim at reconstructing the observations. Another stream of employing auxiliary tasks is proposed to capture predictive information. Self-predictive representations (SPR) (Schwarzer et al. 2021) and prediction of bootstrap latents (PBL) (Guo et al. 2020) predict latent state representation in the future step over a number of steps. Moreover, there are works that primarily revolve around the use of contrastive learning (Stooke et al. 2021; Laskin, Srinivas, and Abbeel 2020; Banino et al. 2022) which leverages the similarity between samples to capture the structure and patterns of data and provides rich and insightful representations. Avoiding extracting extraneous information, state abstraction methods (Zhang et al. 2021a; Kemertas and Aumentado-Armstrong 2021) investigate the similarity between states based on measuring their corresponding rewards and transition probabilities. Nevertheless, the above methods consistently rely on dense reward supervision, rendering them unsuitable for practical scenarios with sparse rewards and vulnerable to instability or collapse. To this end, we focus on the context of uninformative reward signals in procedurally-generated environments.

Preliminaries

Markov Decision Process. The process of an agent who interacts with the environment can be modeled as a discrete-time Markov decision process (MDP), described by a tuple $\langle \mathcal{S}, \mathcal{P}, \mathcal{A}, \mathcal{R}, \rho_0, \gamma \rangle$, specifying the state space $\mathcal{S}$, the transition function $\mathcal{P}(s_{t+1}|s_t, a_t)$, the action space $\mathcal{A}$, the reward function $\mathcal{R}(s_t, a_t)$, initial state distribution ρ_0 and the discount factor $\gamma \in [0, 1)$. In each time step, the agent receives the state $s_t \in \mathcal{S}$ from the environment, then performs the action $a_t \in \mathcal{A}$ according to its policy. The environment transits to the next state according to transition function $\mathcal{P} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$, and the agent receives the next state s_{t+1} and reward $r_t = \mathcal{R}(s_t, a_t)$. The objective of the reinforcement learning agent is to maximize the expected cumulative discounted rewards: $\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, a_t)]$. In this work, we concentrate on environments with sparse extrinsic rewards, where the intrinsic reward is developed to assist the decision-making of the agent. Thus, maximizing $r_t = r_t^e + \beta r_t^i$, where β is the trade-off for balancing extrinsic reward and intrinsic reward, is the optimal policy that the RL agent aims to learn through trial and error.

Bisimulation Metric. The pseudometric $d : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ serves as a measure of equivalence among the states of the MDP. It ensures that states considered equivalent exhibit identical value functions under the policy of the agent. This allows us to partition the state space into equivalence classes and the bisimulation metric measures the "behavioral similarity" between two states based on the reward and dynamics difference.

Definition 1. Given two states s_i, s_j, and an approximate dynamics model $\hat{\mathcal{P}} : \mathcal{S} \times \mathcal{A} \to \mathcal{M}(\mathcal{S}')$, on-policy bisimulation metric with approximate dynamics is:

$$d^\pi(s_i, s_j) = c_R |r_i^\pi - r_j^\pi| + c_T W_1(d)(\hat{\mathcal{P}}^\pi(\cdot|s_i), \hat{\mathcal{P}}^\pi(\cdot|s_j)), \quad (1)$$

where

$$r_i^\pi = \mathbb{E}_{a\sim\pi}[\mathcal{R}(s_i, a)], r_j^\pi = \mathbb{E}_{a\sim\pi}[\mathcal{R}(s_j, a)], \quad (2)$$

$W_1(d)$ is the 1-Wasserstein distance, $c_R \in [0, \infty)$ and $c_T \in [0, 1)$.

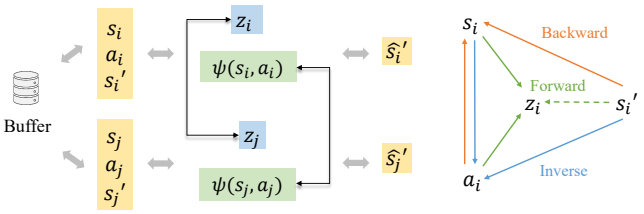
Successor Representation and Successor Feature. The successor representation (Dayan 1993) encapsulates the temporal structure of the Markov decision process (MDP) by capturing the relationships between states and the likelihood of transitioning from one state to another under a specific policy.

$$\Psi^\pi(s, s') = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \mathbb{I}\{s_t = s'\}|s_0 = s\right], \quad (3)$$

where $\mathbb{I}$ denotes the indicator function.

The successor feature (Barreto et al. 2017; Lehnert and Littman 2020) extends the successor representation framework to the context of function approximation, which incorporates the information of the entire trajectory into a single state. Given the embedding ϕ and policy π, the successor feature ψ^π of the state-action pair (s, a) is formulated as follows:

$$\psi_\pi(s, a) = \mathbb{E}_\pi\left[\sum_{t'=t}^{\infty} \gamma^{t'-t} \phi(s_{t'})|s_t = s, a_t = a\right]. \quad (4)$$



(a) Our behavioral metric (b) Cyclic dynamics

Figure 1: (a) Our behavioral metric, through measuring the distance between latent variables and the distance between successor features respectively, accounts for the immediate and cumulative effects simultaneously. (b) The cyclic dynamics is built by integrating our proposed backward dynamics learning, with forward and inverse dynamics.

Assuming $\xi(s, a)$ is the state-action representation function that acts as the basis function to facilitate the reward predictor, w is the weight vector, and the reward predictor is $r(s, a) = \xi(s, a) \cdot \mathrm{w}$, then we have

$$\begin{aligned} Q_\pi(s, a) &= \mathbb{E}_\pi\Big[\sum_{t'=t}^{\infty} \gamma^{t'-t} r(s'_t, a'_t)|s_t = s, a_t = a\Big] \\ &= \mathbb{E}_\pi\Big[\sum_{t'=t}^{\infty} \gamma^{t'-t} \xi(s'_t, a'_t)^T \mathrm{w}|s_t = s, a_t = a\Big] \\ &= \psi_\pi(s, a)^T \mathrm{w}. \end{aligned} \quad (5)$$

Hence, we can conclude that the successor feature $\psi_\pi(s, a)$ possesses the capability to anticipate the cumulative rewards the agent can achieve over time, which serves as a valuable complement to uninformative rewards.

Methodology

To achieve effective abstraction and exploration, we propose a novel *B*ehavioral metric with *C*yclic *D*ynamics (BCD), which is composed of a novel behavioral metric (Figure 1(a)) and cyclic dynamics (Figure 1(b)). Our behavioral metric is designed to capture a robust and informative representation of the state. In addition, cyclic dynamics is established to perceive the complex relationships between states and actions, and then understand the environment more comprehensively. The overall pipeline of BCD is shown as Figure 2.

Behavioral Metric

Behavioral metric-based representation learning is to acquire an embedding space that can sustain behavioral similarity after mapping states onto it. Its objective is shown as follows:

$$\ell(\phi) = \mathbb{E}\left[\left(d(\phi(s_i), \phi(s_j)) - d^\pi(s_i, s_j)\right)^2\right], \quad (6)$$

where $d(\cdot)$ is the distance between two representations of states and $d^\pi(\cdot)$ is the corresponding behavioral metric. $\phi : S \to \mathbb{R}^d$ embeds the state into d-dimensional space. In this

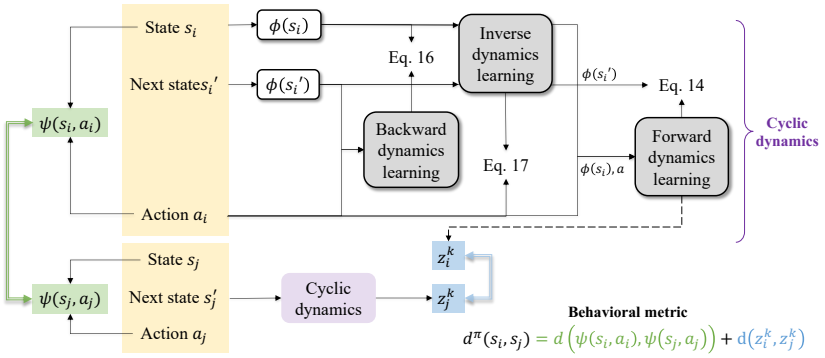


Figure 2: The overall structure of the proposed method BCD. The cyclic dynamics is composed of the backward, forward and inverse dynamics. We employ the latent variable and vector quantization techniques in forward dynamics learning, where the cluster centers are utilized for the transition function estimation in our behavioral metric.

paper, we propose a novel metric and incorporate two perspectives into d^π, the immediate effect and the cumulative effect.

First, we estimate the representation of the next state based on the representation of the state and action, i.e., the transition dynamics, as follows:

$$z = f(\phi(s), a), \tag{7}$$

where f is the transition dynamics predictor. This estimation refers to the prediction of the next step, i.e., one-step prediction, considering the immediate effect. Then, we incorporate this prediction into our behavioral metric, which measures the cosine similarity of the predicted next states:

$$d(f(\phi(s_i), a_i), f(\phi(s_j), a_j)) = \frac{z_i \cdot z_j}{\|z_i\| \|z_j\|}. \tag{8}$$

To be noted, z will be further improved in the next section to achieve the discrete and interpretable goal.

Second, we incorporate the successor feature to serve as the accumulative impact, in the meanwhile making up for the regret of sparse extrinsic reward. The successor feature is a representation of the expected future state occupancy under a certain policy in reinforcement learning, which provides a means to capture the transition dynamics of an environment. It serves as a compressed representation of the dynamics, enabling the agent to plan ahead and make informed decisions. The successor feature can be learned by iteratively updating their estimates based on the transitions observed during interactions with the environment, as shown in the following:

$$\ell_{sf} = \mathbb{E}\left[\left(\phi(s) + \gamma\psi_{\theta^-}(s', a') - \psi_\theta(s, a)\right)^2\right], \tag{9}$$

where ψ is the network of successor feature parameterized by θ, and θ^- is its target version. To this end, an agent can effectively capture the temporal structure of the Markov decision process and gain insights into the long-term consequences of its decisions. After obtaining the successor feature, we leverage this into the successor distance into our behavioral metric, which measures the cosine similarity of the successor features:

$$d(\psi(s_i, a_i), \psi(s_j, a_j)) = \frac{\psi(s_i, a_i) \cdot \psi(s_j, a_j)}{\|\psi(s_i, a_i)\| \|\psi(s_j, a_j)\|}. \tag{10}$$

States that exhibit similar successor features are considered functionally similar, indicating that they are likely to lead to similar long-term outcomes. The objective of our behavioral metric is to minimize the following equation:

$$\ell_{bc}(\phi) = \frac{1}{2}\mathbb{E}\Bigg[\underbrace{\left(d\left(\phi\left(s_i\right), \phi\left(s_j\right)\right)\right.}_{\text{our behavioral metric}} - \underbrace{d\left(f(\phi(s_i), a_i), f(\phi(s_j), a_j)\right)}_{\text{immediate effect measurement}} - \underbrace{\left.d\left(\psi\left(s_i, a_i\right), \psi\left(s_j, a_j\right)\right)\right)}_{\text{cumulative effect measurement}}\Bigg]^2. \tag{11}$$

Cyclic Dynamics

The cyclic dynamics comprises three components, forward, backward and inverse dynamics learning, as depicted in Figure 1(b).

Learning Discrete Latent Variable for Forward Dynamics Learning. In contrast to previous approaches that directly use neural networks to predict the next state, we introduce to estimate the intermediate latent variable in the one-step forward prediction:

$$f(s_{t+1}|s_t, a_t) = \int_{\mathcal{Z}} p(z_t|s_t, a_t) p(s_{t+1}|z_t) d\mu, \tag{12}$$

where latent space $\mathcal{Z}$ is continuous and μ is the Lebesgue measure on $\mathcal{Z}$. To fulfill the goal of discrete and interpretable representations, we process the continuous representations into discrete ones through the vector quantization method (van den Oord, Vinyals, and Kavukcuoglu 2017), enabling enhanced comprehension and analysis of the latent features. Rather than directly quantizing the continuous representation, vector quantization involves a codebook that acts as a dictionary of discrete embeddings:

$$\hat{z} = q(z) := \arg\min_{z^k \in \mathcal{C}} \|z_e - z^k\|_2, \tag{13}$$

where $\mathcal{C} \in \{z^1, z^2, \ldots, z^K\}$ is the codebook which consists of K embedding vectors and $z^e = f_{enc}(\phi(s), a)$ in which f_{enc} is the encoder network. The encoder output is compared with the embedding vectors in the codebook, and the closest

match is selected as the discrete code representing the input. This quantization step makes the model more interpretable and aids in disentangling the learned representations, which can be viewed as a classical method akin to learning K cluster centers using k-means. Afterward, the selected discrete code is then passed through the decoder, which reconstructs the representation based on the latent variable. The non-differentiable quantization operation is handled through a straight-through gradient estimator, allowing gradients to flow from the decoder to the encoder. To this end, we construct the loss function for learning discrete latent variable for forward dynamics learning as follows:

$$\begin{aligned}\ell_{for} = &\log p\big(\phi(s')|f_{dec}(\hat{z})\big) \\ &+ \|z^k - \text{sg}(z_e)\|_2^2 + \beta\|\text{sg}(z^k) - z_e\|_2^2,\end{aligned} \tag{14}$$

where f_{dec} is the decoder network, and $\text{sg}(\cdot)$ is the stop gradient function. The first component is the reconstruction loss of the decoder, while the second component moves z^k towards the output of the encoder z_e. To promote the commitment of the encoder to a particular embedding and prevent uncontrolled growth of its output, a commitment loss is introduced as a regularization term, which is the third component. By incorporating the vector quantization step, vector quantization encourages the model to capture meaningful and compact representations.

Based on the discrete latent variable z^k obtained, we inject this into our behavioral metric to observe the immediate effect. Instead of measuring the distance between the predicted representation of the next state, we turn to the similarity metric on the cluster center in the discrete space. This enables us to quantify the resemblance between states in a discrete manner and capture the inherent structure of the state space. In this way, the distance between the latent representations which improves Eq. 8 is displayed as follows:

$$d(f(\phi(s_i), a_i), f(\phi(s_j), a_j)) = \frac{z_i^k \cdot z_j^k}{\|z_i^k\|\|z_j^k\|}. \tag{15}$$

States with similar behavioral dynamics exhibit clusters that are closer together in the representation space, while states with distinct behavioral characteristics exhibit clusters that are farther apart. This enables improved interpretability of our behavioral metric and enhances the ability of the agent to capture implicit information.

Backward and Inverse Dynamics Learning. As a supplement to forward dynamics learning, backward dynamics learning involves predicting the representation of the current state based on the representation of the next state and action. Formally, the mean squared error loss is applied to minimize the prediction loss as follows:

$$\ell_{back} = \frac{1}{2}\|f_b(\phi(s_i'), a) - \phi(s_i)\|^2 = \frac{1}{2}\|f_b(z_i', a) - z_i\|^2, \tag{16}$$

where $f_b(\cdot)$ is the backward dynamics predictor. The idea behind this is to capture the semantic relationships between the representations of the states and actions. Through anticipating backward dynamics transitions, valuable contextual information is captured while mitigating the risk of overemphasizing the prediction of forward transitions, thereby enabling the agent to build a more thorough awareness of the environment and make promising decisions.

Algorithm 1: Behavioral metric with cyclic dynamics

```
Initialize the replay buffer D.
Initialize the number of epochs N.
for epoch i=0 to N do
    Get the initial state s_0 from the environment;
    for step t = 0 to terminal do
        Get the representation of state φ(s_t);
        Execute the action a_t based on the policy π(a_t|φ(s_t))
        and obtain the next state s_{t+1} from the environment;
        Estimate the forward dynamics function f(·|φ(s_t), a_t)
        and its corresponding latent variable z_k;
        Estimate the backward dynamics function
        f_b(·|φ(s_{t+1}), a_t);
        Estimate the inverse dynamics function
        g(φ(s_t), φ(s_{t+1}));
        Record data: D ← D ∪ (s_t, a_t, r_t, s_{t+1});
    end for
    for training step τ = 0 to terminal do
        Sample a batch of training data;
        Update the learning of the successor feature (Eq. 9);
        Optimize the overall objective of BCD (Eq. 19);
    end for
end for
```

As another supplement to forward dynamics learning, inverse dynamics learning involves predicting the action based on the representation of the state and the next state:

$$\ell_{inverse} = \frac{1}{2}\|g(\phi(s_i), \phi(s_i')) - a\|^2 = \frac{1}{2}\|g(z_i, z_i') - a\|^2, \tag{17}$$

where $g(\cdot)$ is the inverse dynamics predictor. The motivation behind introducing the mapping is to selectively discard redundant or irrelevant information pertaining to the environment, thereby focusing solely on the aspects that are valuable for accurately predicting the actions of the agent.

Overall, the cyclic dynamics is established, and its loss is as follows:

$$\ell_{cycle} = \ell_{for} + \ell_{back} + \ell_{inverse}. \tag{18}$$

To this end, the overall objective of the proposed BCD is:

$$\min \left[-\mathbb{E}_\pi \left[\sum_t r_t \right] + \ell_{cycle} + \ell_{bc} \right]. \tag{19}$$

Experiments and Analysis

Experimental Setup

Environment. We evaluate on challenging procedurally-generated environment Minigrid (Chevalier-Boisvert et al. 2023). The grids consist of either a single object or none, and may include various object types, such as walls, doors, keys, balls, boxes, and goals, with distinct colors representing each category. The specific objective of the domains ranges from retrieving a key to matching objects of similar colors. The agent navigates through the grid, avoids obstacles, and collects objects in order to reach the goal state. The reward structure in Minigrid is typically sparse, where the agent receives a sparse reward signal of 1 upon successfully reaching the destination, while a 0 for failure.

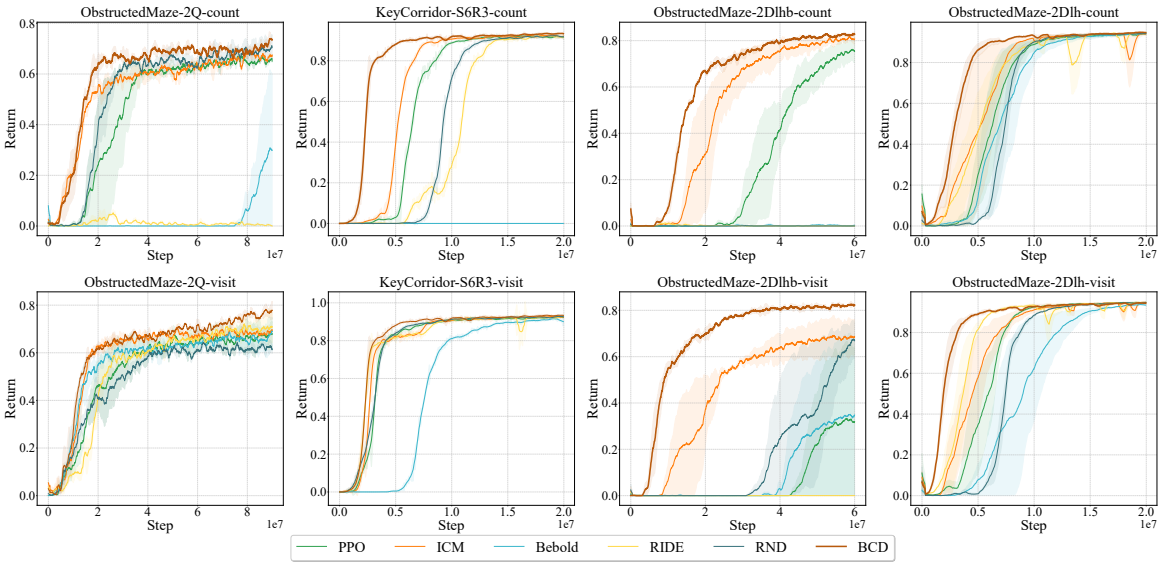


Figure 3: Comparisons with state-of-the-art methods.

Taking the environment MiniGrid-ObstructedMaze-2Q as an example, the observation is an egocentric view of the environment which is encoded as a 3-dimensional tuple (object_id, color_id, state), where state refers to the door state with 0 for open, 1 for closed, and 2 for locked. There are seven actions available for the agent: turn left, turn right, move forward, pick up an object, drop an object, toggle and done.

Baselines and Implement Details. We compare our proposed method with some baselines in the MiniGrid environment: (1) PPO (vanilla) (Schulman et al. 2017), (2) ICM (Pathak et al. 2017), (3) Bebold (Zhang et al. 2020); (4) RIDE (Raileanu and Rocktäschel 2020) and (5) RND (Burda et al. 2019). All these baselines will be evaluated utilizing the visit-based and count-based intrinsic rewards and PPO is the base RL algorithm in our study, following the setting in (Wang et al. 2023). The convolutional neural network is employed to deal with the input observation and the coefficient β for intrinsic reward is the same setting as (Wang et al. 2023) for the fair comparison. Our code is available at https://github.com/AnneZhu1020/BCD.

Results and Analysis

Comparison with State-of-the-Arts. To validate the effectiveness of our proposed method, we conduct a comprehensive comparison with state-of-the-art methods, as presented in Figure 3. We plot the mean results for 5 different seeds and the shadow areas present their variance. Our proposed BCD outperforms previous methods with respective count-based episodic reward and visit-based episodic reward. Specifically, BCD in ObstructedMaze-2Q which is a hard exploration environment surpasses others by a good margin, demonstrating its effectiveness. Additionally, BCD achieves faster convergence than others on the series environments of KeyCorridor and MultiRoom, demonstrating its efficiency. More results are displayed in Appendix of our full version online at https://github.com/AnneZhu1020/BCD. The above results provide strong evidence of the superior performance and efficiency of our proposed method across various challenging environments.

Method	ObstructedMaze-2Q	KeyCorridor-S6R3
×forward dyn. learn.	0.77 ± 0.245	0.94 ± 0.039
×backward dyn. learn.	0.74 ± 0.288	0.93 ± 0.021
BCD	0.78 ± 0.182	0.94 ± 0.012

Table 1: Ablation studies.

Ablation Studies. To investigate the contribution of our forward and backward dynamics leanings, we conduct ablation studies on them and report the results in Table 1. The mean and variance across 5 seeds in the environments ObstructedMaze-2Q and KeyCorridor-S6R3 are recorded. In the absence of forward dynamics learning, the model exhibits higher variance, and without the backward dynamics learning component, there is a slight degradation in performance. These findings underscore the significance of both components in ensuring the stability of our proposed method, ultimately enhancing its performance. More ablation studies are illustrated in Appendix.

Effect of Codebook Size and Code Length. We investigate how the number of codes in the codebook and its corresponding length affect our proposed method. The number of codes is tested in the range of $\{4, 8, 16\}$, and the length of the code is tested in the range of $\{16, 32, 64\}$. It can be observed from Figure 5 that the variation in the numbers of code and code lengths have a slight impact on the final results. We speculate that this effect is related to the complex-

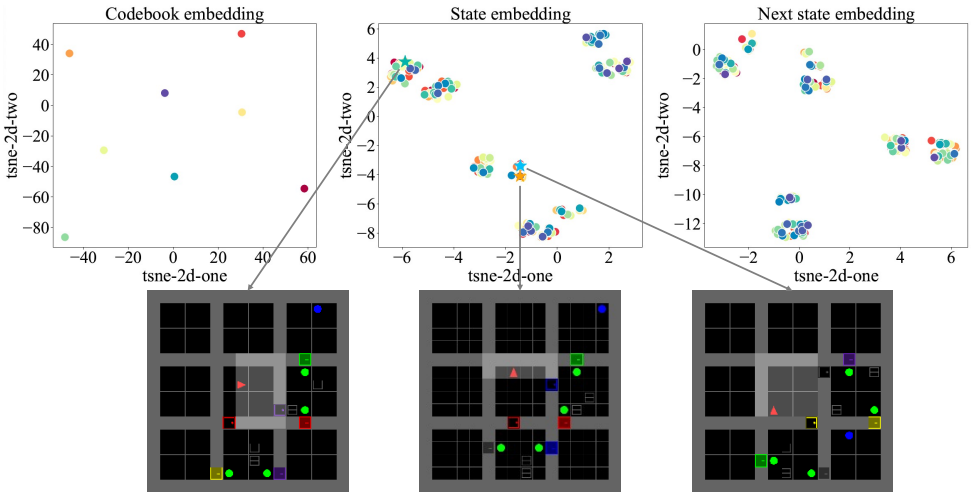


Figure 4: t-SNE visualization of representations of code, state embedding, and next state embedding.

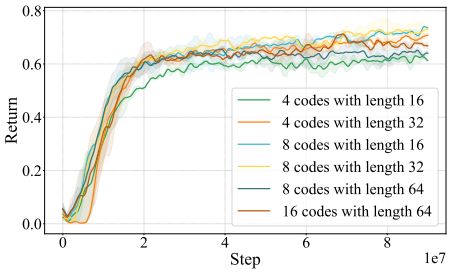


Figure 5: Effect of codebook size and code length.

ity of the environment. In highly complex environments, it may be necessary to use a larger number of codes to accurately model the state information where insufficient codes may result in semantic overlap and inadequate representation of the environment. Conversely, in relatively simple environments, an excessive number of codes may introduce interference and hinder the precise expression of the codes, making it challenging for the agent to accurately classify the current state into the appropriate cluster. Overall, the flexibility and robustness of our proposed method mitigate the potential challenges posed by changes in the number of codes and code length, allowing for effective and reliable performance in a variety of scenarios.

Visualization of Representations. For a better understanding of learned embedding and visualization, we employ t-SNE (van der Maaten and Hinton 2008) to project the representations onto a 2-dimensional space. As shown in Figure 4, we evaluate the trained model on 256 simulated environments and plot the corresponding codebook of its policy, state and next state embedding respectively. The left of Figure 4 illustrates the codebook of the policy, where the 8 codes are perfectly distributed among 8 distinct points and the clustering results demonstrate the effectiveness of the embedding learning. The middle and right of Figure 4 display the embeddings of the states and next states respectively. It can be observed that they also cluster perfectly into 8 distinct clusters, highlighting the effectiveness of the discrete clustering in the state embeddings. Additionally, we provide representations of the current states for three of the environments. It is evident that the environments marked with orange and blue stars, belonging to the same cluster, exhibit behavioral similarities in their states. On the other hand, the environment marked with a green star shows different behavioral similarities compared with the red and yellow star-marked environments. This indicates that our method can discern behavioral similarities among agent states at a fine granularity, showcasing the effectiveness of our proposed metric and representation learning.

Conclusion

We identify the challenges faced by existing methods in procedurally-generated environments, particularly concerning uninformative extrinsic reward and high stochasticity, leading to representation collapse and insufficient interrelation among dynamics. To overcome these limitations, we provide a novel behavioral metric that effectively captures both immediate and cumulative effects, promoting the acquisition of robust and informative representation. Specifically, our behavioral metric leverages the successor feature, encapsulating the expected future cumulative rewards to break down the dependency on reward dynamics. For transition dynamics, we leverage the latent variable and vector quantization to evaluate the distance between their clusters. The cyclic dynamics is established that incorporates our proposed backward dynamics learning with forward and inverse dynamics, allowing the agent to develop a more thorough awareness of the environmental dynamics. Our novel behavioral metric represents a promising approach to tackle the challenges posed by procedurally-generated environments and provides a new insight to pave the way for further advancements in reinforcement learning.

Acknowledgements

This work was partially supported by the National Natural Science Foundation of China (No. 62276047) and Australian Research Council (No. DP190102353 and No. CE200100025).

References

Afsar, M. M.; Crump, T.; and Far, B. H. 2023. Reinforcement Learning based Recommender Systems: A Survey. *ACM Comput. Surv.*, 55(7): 145:1–145:38.

Badia, A. P.; Sprechmann, P.; Vitvitskyi, A.; Guo, Z. D.; Piot, B.; Kapturowski, S.; Tieleman, O.; Arjovsky, M.; Pritzel, A.; Bolt, A.; and Blundell, C. 2020. Never Give Up: Learning Directed Exploration Strategies. In *8th International Conference on Learning Representations, ICLR, Addis Ababa, Ethiopia, April 26-30*.

Bai, C.; Liu, P.; Liu, K.; Wang, L.; Zhao, Y.; Han, L.; and Wang, Z. 2023. Variational Dynamic for Self-Supervised Exploration in Deep Reinforcement Learning. *IEEE Trans. Neural Networks Learn. Syst.*, 34(8): 4776–4790.

Banino, A.; Badia, A. P.; Walker, J. C.; Scholtes, T.; Mitrovic, J.; and Blundell, C. 2022. CoBERL: Contrastive BERT for Reinforcement Learning. In *The Tenth International Conference on Learning Representations, ICLR, Virtual Event, April 25-29*.

Barreto, A.; Dabney, W.; Munos, R.; Hunt, J. J.; Schaul, T.; Silver, D.; and van Hasselt, H. 2017. Successor Features for Transfer in Reinforcement Learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, December 4-9, Long Beach, CA, USA*, 4055–4065.

Burda, Y.; Edwards, H.; Storkey, A. J.; and Klimov, O. 2019. Exploration by random network distillation. In *7th International Conference on Learning Representations, ICLR, New Orleans, LA, USA, May 6-9*.

Chevalier-Boisvert, M.; Dai, B.; Towers, M.; de Lazcano, R.; Willems, L.; Lahlou, S.; Pal, S.; Castro, P. S.; and Terry, J. 2023. Minigrid & Miniworld: Modular & Customizable Reinforcement Learning Environments for Goal-Oriented Tasks. *CoRR*, abs/2306.13831.

Dayan, P. 1993. Improving Generalization for Temporal Difference Learning: The Successor Representation. *Neural Comput.*, 5(4): 613–624.

Flet-Berliac, Y.; Ferret, J.; Pietquin, O.; Preux, P.; and Geist, M. 2021. Adversarially Guided Actor-Critic. In *9th International Conference on Learning Representations, ICLR, Virtual Event, Austria, May 3-7*.

Guo, Z. D.; Pires, B. Á.; Piot, B.; Grill, J.; Altché, F.; Munos, R.; and Azar, M. G. 2020. Bootstrap Latent-Predictive Representations for Multitask Reinforcement Learning. In *Proceedings of the 37th International Conference on Machine Learning, ICML, 13-18 July, Virtual Event*, 3875–3886.

Hafner, D.; Lillicrap, T. P.; Fischer, I.; Villegas, R.; Ha, D.; Lee, H.; and Davidson, J. 2019. Learning Latent Dynamics for Planning from Pixels. In *Proceedings of the 36th International Conference on Machine Learning, ICML, 9-15 June, Long Beach, California, USA*, 2555–2565.

Kemertas, M.; and Aumentado-Armstrong, T. 2021. Towards Robust Bisimulation Metric Learning. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-14, virtual*, 4764–4777.

Kendall, A.; Hawke, J.; Janz, D.; Mazur, P.; Reda, D.; Allen, J.; Lam, V.; Bewley, A.; and Shah, A. 2019. Learning to Drive in a Day. In *International Conference on Robotics and Automation, ICRA, Montreal, QC, Canada, May 20-24*, 8248–8254.

Lai, T. L.; and Robbins, H. 1985. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1): 4–22.

Laskin, M.; Srinivas, A.; and Abbeel, P. 2020. CURL: Contrastive Unsupervised Representations for Reinforcement Learning. In *Proceedings of the 37th International Conference on Machine Learning, ICML, 13-18 July, Virtual Event*, 5639–5650.

Lehnert, L.; and Littman, M. L. 2020. Successor Features Combine Elements of Model-Free and Model-based Reinforcement Learning. *J. Mach. Learn. Res.*, 21: 196:1–196:53.

Levine, S.; Finn, C.; Darrell, T.; and Abbeel, P. 2016. End-to-End Training of Deep Visuomotor Policies. *J. Mach. Learn. Res.*, 17: 39:1–39:40.

Moerland, T. M.; Broekens, J.; and Jonker, C. M. 2017. Efficient exploration with Double Uncertain Value Networks. *CoRR*, abs/1711.10789.

Osband, I.; Roy, B. V.; Russo, D. J.; and Wen, Z. 2019. Deep Exploration via Randomized Value Functions. *J. Mach. Learn. Res.*, 20: 124:1–124:62.

Pathak, D.; Agrawal, P.; Efros, A. A.; and Darrell, T. 2017. Curiosity-Driven Exploration by Self-Supervised Prediction. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops, Honolulu, HI, USA, July 21-26*, 488–489.

Raileanu, R.; and Rocktäschel, T. 2020. RIDE: Rewarding Impact-Driven Exploration for Procedurally-Generated Environments. In *8th International Conference on Learning Representations, ICLR, Addis Ababa, Ethiopia, April 26-30*.

Schrittwieser, J.; Antonoglou, I.; Hubert, T.; Simonyan, K.; Sifre, L.; Schmitt, S.; Guez, A.; Lockhart, E.; Hassabis, D.; Graepel, T.; Lillicrap, T. P.; and Silver, D. 2020. Mastering Atari, Go, chess and shogi by planning with a learned model. *Nat.*, 588(7839): 604–609.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal Policy Optimization Algorithms. *CoRR*, abs/1707.06347.

Schwarzer, M.; Anand, A.; Goel, R.; Hjelm, R. D.; Courville, A. C.; and Bachman, P. 2021. Data-Efficient Reinforcement Learning with Self-Predictive Representations. In *9th International Conference on Learning Representations, ICLR, Virtual Event, Austria, May 3-7*.

Stooke, A.; Lee, K.; Abbeel, P.; and Laskin, M. 2021. Decoupling Representation Learning from Reinforcement

Learning. In *Proceedings of the 38th International Conference on Machine Learning, ICML, 18-24 July, Virtual Event*, 9870–9879.

van den Oord, A.; Vinyals, O.; and Kavukcuoglu, K. 2017. Neural Discrete Representation Learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, December 4-9, Long Beach, CA, USA*, 6306–6315.

van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *J. Mach. Learn. Res.*, 9: 2579–2605.

Wang, K.; Zhou, K.; Kang, B.; Feng, J.; and Yan, S. 2023. Revisiting Intrinsic Reward for Exploration in Procedurally Generated Environments. In *The Eleventh International Conference on Learning Representations, ICLR, Kigali, Rwanda, May 1-5*.

Yarats, D.; Zhang, A.; Kostrikov, I.; Amos, B.; Pineau, J.; and Fergus, R. 2021. Improving Sample Efficiency in Model-Free Reinforcement Learning from Images. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI, Virtual Event, February 2-9*, 10674–10681.

Zhang, A.; McAllister, R. T.; Calandra, R.; Gal, Y.; and Levine, S. 2021a. Learning Invariant Representations for Reinforcement Learning without Reconstruction. In *9th International Conference on Learning Representations, ICLR, Virtual Event, Austria, May 3-7*.

Zhang, T.; Xu, H.; Wang, X.; Wu, Y.; Keutzer, K.; Gonzalez, J. E.; and Tian, Y. 2020. BeBold: Exploration Beyond the Boundary of Explored Regions. *CoRR*, abs/2012.08621.

Zhang, T.; Xu, H.; Wang, X.; Wu, Y.; Keutzer, K.; Gonzalez, J. E.; and Tian, Y. 2021b. NovelD: A Simple yet Effective Exploration Criterion. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS 2021, December 6-14, virtual*, 25217–25230.