AUTHOR QUERIES

AUTHOR PLEASE ANSWER ALL QUERIES

- **Your proof has been created based on Society and IEEE layout and editing requirements.**
- **Carefully check the page proofs (and coordinate with all authors). Please check author names and affiliations, funding, as well as the overall article for any errors prior to sending in your author proof corrections.**
- **Additional changes or updates WILL NOT be accepted after the article is published online/print in its final form.**
- **New source files WILL NOT be accepted at this time.**
- **Once the article is finalized, you will receive an invoice for any applicable charges from the Copyright Clearance Center (CCC). Please visit https://journals.ieeeauthorcenter.ieee.org/your-role-in-article-production/about-potential-article-processing-charges/ for a comprehensive list of article charges. If you have any questions regarding overlength page charges, need an invoice, or have any other billing questions, please contact apcinquiries@ieee.org.**

AQ:1 = Please confirm or add details for any funding or financial support for the research of this article.

AQ:2 = Please provide the postal code for Chinese Academy of Sciences and University of Chinese Academy of Sciences.

AQ:3 = Please confirm whether the edits made in the current affiliation of all the authors are correct.

AQ:4 = Please provide the location (city, country) and postal code for State Key Laboratory of Brain Cognition and Brain-Inspired Intelligence Technology.

AQ:5 = Please provide the appropriate section number for the phrase "following sections" throughout the article.

AQ:6 = Please provide the descriptions of part labels (a)–(c) in Fig. 8 and (a)'(e) in Fig. 11.

AQ:7 = Please provide the page range for Ref. [8].

AQ:8 = Please provide the month, page range, and volume no. for Ref. [31].

Learning Stable Subgoal Representation in Hierarchical Reinforcement Learning With Causal Factorization

Yibo Zhang, Dengpeng Xing, and Tielin Zhang

***Abstract*—Goal-conditioned hierarchical reinforcement learning (GCHRL) has proven effective for complex control tasks, particularly in long-horizon and sparse-reward settings. By employing a multilevel policy hierarchy, the high-level policy generates feasible subgoals for the low-level policy, which in turn learns to achieve them. This structure enhances exploration and substantially improves learning efficiency. However, the subgoal space has a profound influence on policy training. A well-constructed subgoal space is essential for efficient learning, while a poorly learned one hinders it. Causal reinforcement learning (CRL), which integrates causal learning into reinforcement learning, enhances decision-making through discovery and utilization of causal relations. In this article, inspired by current CRL methods, we propose a novel causal approach to learning subgoal spaces. Leveraging causal relations among different parts of the state space, we extract components containing more high-level and decision-relevant information. Besides, to enhance the quality of representation learning, we propose a regularization method to alleviate representation collapse and improve stability. We evaluate our approach on a series of long-horizon and sparse-reward tasks. The experimental results show that our method learns higher quality subgoal representations and outperforms current methods.**

***Index Terms*—Causal factorization, hierarchical reinforcement learning (HRL), representation learning.**

I. Introduction

HIERARCHICAL reinforcement learning (HRL) [1] has demonstrated effectiveness in addressing a wide range of challenging tasks, including long-horizon control tasks and games with sparse rewards [2], [3], [4], [5], significantly outperforming standard deep reinforcement learning (DRL) [6] methods. Among the predominant HRL paradigms, goal-conditioned HRL (GCHRL) is particularly promising for complicated goal-reaching tasks [7], [8], [9]. This capability enhancement stems from GCHRL's hierarchical structure, wherein high-level and low-level policies operate at different temporal scales. The high-level policy sets subgoals periodically, and the low-level policy executes a series of primitive actions to achieve them. This framework enables the high-level policy to decompose the task into a series of manageable subtasks, thereby allowing the low-level policy to focus solely on reaching the assigned subgoals. A critical component bridging the gap between the two layers is the subgoal space [10], which serves as the action space of the high-level policy and the goal space of the low-level policy. A well-learned subgoal space can accelerate the training process. Moreover, as the subgoal space usually has lower dimensionality, it reduces the complexity of subgoal selection, mitigating the curse of dimensionality.

Causal reinforcement learning (CRL) [11], which combines causal learning [12] and DRL, has achieved remarkable advances in data efficiency, interpretability, and robustness [13], [14]. Leveraging causal inference and causal graph models (CGMs), CRL agents can uncover underlying causal mechanisms within observed data, such as causal relations among variables or low-dimensional causal representations. This causal understanding enables agents to mitigate the influence of spurious correlations, allowing them to extract more relevant information from observations and improve learning efficiency. Specifically, planning with sparse causal dynamic models leads to improved performance and generalization [13], [15], while training agents in causal representation spaces increases their robustness against environmental distractions [16].

Inspired by current CRL schemes, we propose a novel subgoal representation learning approach through causal factorization in the latent space. Mainstream causal representation learning methods learn a map from the original state space to a more compact latent representation space, preserving key information for decision-making and discarding most irrelevant parts. Different from these methods, we take another approach. We combine causal dynamic learning and representation learning, integrating CGMs into the representation learning process. Specifically, we assume a simplified latent causal structure and decompose the latent space into causal and noncausal factors. We then model the causal relations between these factors and learn a factored latent dynamic transition

AQ:1 Received 10 November 2025; revised 24 May 2026; accepted 23 August 2026. This work was supported by the Strategic Priority Research Program AQ:2 of Chinese Academy of Sciences under Grant XDB1010302. *(Corresponding authors: Dengpeng Xing; Tielin Zhang.)*

AQ:3 Yibo Zhang is with the Institute of Automation, Chinese Academy of Sciences, Beijing, China, and also with the Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences, Shanghai, China (e-mail: zhangyibo2023@ia.ac.cn).

Dengpeng Xing is with the Institute of Automation, Chinese Academy of Sciences, Beijing, China, and also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China (e-mail: dengpeng.xing@ia.ac.cn).

Tielin Zhang is with the Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences, Shanghai, China, and AQ:4 also with the State Key Laboratory of Brain Cognition and Brain-Inspired Intelligence Technology (e-mail: zhangtielin@ion.ac.cn).

Digital Object Identifier 10.1109/TNNLS.2026.3729654

model that conforms to this causal structure. This results in a causal representation space enriched with higher level, decision-relevant information, which serves as the subgoal representation space in our GCHRL framework. In addition, to alleviate representation collapse and stabilize the learned space, we introduce a regularization loss that jointly considers both of these aspects. Built upon the well-learned subgoal space, we deploy a hybrid exploration strategy to assist policy learning.

In summary, our main contributions are as follows.

1) We propose a novel subgoal representation learning method using causal factorization to extract decision-relevant information from the state space, yielding a higher quality subgoal space that enhances policy learning efficiency.
2) We introduce a regularization technique to mitigate representation collapse and stabilize the learned subgoal space, thereby facilitating more robust policy training.
3) Based on the well-learned subgoal space, we develop a hybrid strategy combining random network distillation (RND) [17] and count-based methods to enhance exploration.

II. Related Work

A. Goal-Conditioned HRL

GCHRL framework is designed to address various complex goal-reaching tasks. Leveraging a hierarchical policy structure, GCHRL agents have enhanced capabilities in exploration and exploitation. However, training both levels of policies simultaneously can lead to instability issues. Prior works tried to mitigate this issue primarily through off-policy correction [18] or hindsight replay [19], both of which have been proven effective in stabilizing the training dynamics. Built upon stable dynamics, a variety of methods have been proposed to enhance policy learning. Some works focus on exploration by encouraging agents to explore novel states [7], [20], while others emphasize exploitation by modeling subgoal reachability explicitly or implicitly [8], [21], [22].

B. Subgoal Representation Learning in GCHRL

In GCHRL, the subgoal space serves as a bridge between the high- and low-level policies, making it essential to overall performance. Early methods [3] use the entire state space as the subgoal space, but when facing high-dimensional tasks, it becomes infeasible due to "dimensionality explosion." To address this issue, some approaches adopt predefined subgoal spaces, such as the xy space in navigation or manipulation tasks [21], [23]. However, this approach is limited when facing unknown or unstructured state spaces. Alternatively, several works have explored learning subgoal spaces via representation learning. For instance, representations based on states with slower dynamics have been proposed to facilitate more efficient exploration [10]. To improve stability and generalization to unseen states, probabilistic modeling techniques are further introduced [24]. Furthermore, to mitigate representation drift during training, penalties on changes between old and new representations are incorporated [7]. However, there still remain some challenges that current approaches cannot adequately address, such as representation collapse and limited interpretability. Regarding these issues, we propose to take advantage of causal modeling. Leveraging global causal relations, representation collapse can be alleviated, and interpretability can be enhanced.

C. Causality in Reinforcement Learning

Integrating causality into DRL has been shown to be beneficial. DRL agents with causal knowledge gain higher data efficiency and better generalization. Broadly, causality can be incorporated into DRL through two main avenues: causal model-based DRL and causal representation learning. Causal model-based DRL emphasizes explicit causal discovery and modeling [13], [25], wherein agents learn dynamic models with sparse causal structures, as opposed to previous dense ones. This helps mitigate the influence of spurious correlations in the data and improves generalization to out-of-distribution states [11]. Causal representation learning focuses on learning state representations under causal constraints [16], [26]. By preserving information about underlying causal mechanisms and filtering out redundant features, this enables agents to focus more on causally meaningful aspects of the environment.

III. Preliminaries

A. Framework of GCHRL

A GCHRL problem is usually modeled as a semi-Markov decision process (SMDP), represented by a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma, \mathcal{G})$, where $\mathcal{S}, \mathcal{A}, \mathcal{G}$ denote the state space, action space, and goal space, respectively. $\mathcal{P} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ represents the state transition dynamics, $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ represents the reward function, and γ is the discount factor. Besides, the high- and low-level policy can be represented as $\pi_h(sg|s, g; \theta_h)$ and $\pi_l(a|s, sg; \theta_l)$, respectively, where θ_h, θ_l denote parameters of corresponding networks, and $sg \in \mathcal{SG}$ is the subgoal in the subgoal space $\mathcal{SG}$.

The high-level policy operates at a lower temporal resolution, issuing a new subgoal every H steps. In one episode, the subgoal sg_t at timestep t is sampled from the high-level policy, $sg_t \sim \pi_h(sg|s_t, g; \theta_h)$ when $t \equiv 0 (\text{mod } H)$. Between high-level decisions, sg_t is determined by subgoal transition $h(sg_{t-1}, s_{t-1}, s_t)$, e.g., $sg_t = sg_{t-1} + \phi(s_{t-1}) - \phi(s_t)$ for relative subgoals with any subgoal representation function $\phi : \mathcal{S} \to \mathcal{SG}$ and $sg_t = sg_{t-1}$ for absolute subgoals. At each timestep t, the low-level action a_t is sampled by $a_t \sim \pi_l(a|s_t, sg_t; \theta_l)$. The high-level reward is typically defined as cumulative external rewards over H steps, $r_{h,t} = \sum_{i=0}^{H-1} r_{t+i}$, and the high-level policy aims to maximize the expected discounted return, i.e., $\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_{h,Ht}\right]$. The low-level reward incentivizes reaching the current subgoal, commonly defined as $r_{l,t} = -\|\phi(s_{t+1}) - sg_t\|_2$, where ϕ is the subgoal representation function. Both levels of policies can be trained with standard DRL algorithms.

B. Causal Reinforcement Learning

Causality has been integrated into RL primarily through two avenues: CGMs and causal representation learning.

1) Causal Graph Models: We denote the state space as $\mathcal{S} = \{S_1, \ldots, S_n\}$ and the action space as $\mathcal{A} = \{A_1, \ldots, A_m\}$, where uppercase letters denote random variables, and n and m are the numbers of state and action variables, respectively. We use lowercase letters $s = \{s_1, \ldots, s_n\}$ and $a = \{a_1, \ldots, a_m\}$ to represent specific states and actions.

In the context of CRL, the state transition dynamics are typically modeled as a CGM, also known as a factored Markov decision process (FMDP) [27] or a dynamic Bayesian network (DBN) [28] in RL settings. Causality depicts the influence relationships among state variables, or between state and action variables.

Definition 1 (Causal Graph Model): Consider the current state S, action A, and the next state S'. A CGM is represented by a tuple $(\mathcal{G}, P)$, where $\mathcal{G}$ is a directed acyclic causal graph on S, A, S' specifying the parent set $Pa(S'_i)$ for each next state $S'_i \in S'$, and P is the corresponding transition distribution. The transition dynamics can be written as

$$P\left(S' \middle| S, A\right) = \prod_{i=1}^{n} P\left[S'_i \middle| Pa\left(S'_i\right)\right]. \tag{1}$$

Usually the causal graph $\mathcal{G}$ is unknown and can be learned from data through causal discovery methods.

Proposition 1 (Causal Discovery for CGMs): Assuming that state variables transit independently, i.e., no causal relations between S'_i and S'_j for any $i \neq j$, then the transition model can be represented as a CGM with a bipartite causal graph: all edges start from (S, A) and end in S'. If P is faithful to the dynamics, then $\mathcal{G}$ can be identified by

$$X_i \in Pa\left(S'_j\right) \Leftrightarrow \neg\left[X_i \perp\!\!\!\perp_P S'_j \middle| (S, A) \setminus \{X_i\}\right] \tag{2}$$

where $X_i \in (S, A), S'_j \in S'$, and "$\perp\!\!\!\perp_P$" denotes conditional independence under P.

The proof can be found in [25]. Conditional independence relationships within a causal graph can be identified via various methods, such as fast CIT [29] or conditional mutual information (CMI). Once learned, the CGM can be applied to various downstream tasks, such as model-based planning or data augmentation.

2) Causal Representation Learning: Causal representation learning is a subfield of representation learning. Similar to mainstream approaches, it learns a mapping function $\phi : \mathcal{S} \to \mathcal{Z}$ that projects the original state space into a low-dimensional latent space. However, the key distinction lies in the constraints imposed on the latent space. In RL's context, commonly adopted constraints include cross-domain invariance [30], disentanglement [31], and dynamic consistency [26].

The learned causal representation space can serve as a new, more compact state space for the agent. At timestep t, the agent observes s_t, maps it to $z_t = \phi(s_t)$, and selects an action $a_t \sim \pi(z_t)$. As the z space is typically lower dimensional and more structured than $\mathcal{S}$, the agent can learn an effective policy with less data, enhancing data efficiency.

IV. Methods

Causality has been widely incorporated into DRL, where it has demonstrated its ability to capture general global mechanisms that facilitate improved policy learning. However, its application within GCHRL remains underexplored. Motivated by this characteristic of causality, we propose a novel subgoal representation learning method that constructs the subgoal space from a causal perspective to mitigate representation collapse. Based on this foundation, we further introduce a regularization mechanism to improve stability and alleviate representation collapse and drift. Leveraging the well-learned subgoal space, we further incorporate RND [17] and count-based methods [7], [32] to develop a hybrid exploration strategy within the learned representation space.

We present our **c**ausally factorized subgoal **r**epresentation (CSR) framework, as illustrated in Fig. 1. CSR comprises three components: 1) a primary representation learning module based on causal factorization; 2) a regularization module; and 3) a hybrid exploration module. Each component is detailed in the following sections.

A. Causally Factorized Representation Learning

In CRL, it is widely acknowledged that there exist causal relations among different variables, and not all variables contribute equally to decision-making. For example, VISA [33] decomposes variables into hierarchies according to their causal dependencies, and CDL [13] extracts compact state representation spaces by discarding part of action-irrelevant variables.

Inspired by the aforementioned approaches, we propose a causal scheme for learning subgoal spaces. Different from prior work that focuses on identifying causal relations in the state space, our objective is to learn a structured latent space $\mathcal{Z}$, tailored for GCHRL. To this end, we partition the latent space into two parts: the causal part $\mathcal{Z}_c$, which contains decision-relevant features, and the noncausal part $\mathcal{Z}_{nc}$, which captures auxiliary or irrelevant information. Then, we define a CGM $(\mathcal{G}_z, \mathcal{P}_z)$ on the latent space $\mathcal{Z}$ instead of the original state space $\mathcal{S}$. By definition 1, we denote $Z = \{Z_c, Z_{nc}\}$ as the latent variables, and $Z' = \{Z'_c, Z'_{nc}\}$ as their successors at the next timestep. As we focus on learning state representations, the original action variable A is retained. $\mathcal{G}_z$ is a causal graph on Z, A, Z', and P_z is the latent transition distribution. Since causal discovery in a learnable latent space is inherently underdetermined, we adopt a modeling assumption to guide this process. In particular, we impose causal constraints $Pa(Z'_c) = \{Z_c, Z_{nc}, A\}$ and $Pa(Z'_{nc}) = \{Z_{nc}, A\}$. Then, P_z can be factorized as

$$P\left(Z' \middle| Z, A\right) = P\left(Z'_c \middle| Z_c, Z_{nc}, A\right) \cdot P\left(Z'_{nc} \middle| Z_{nc}, A\right). \tag{3}$$

This factorization reflects the causal structure of the latent space $\mathcal{Z}$.

In GCHRL, an effective subgoal space should distill essential low-dimensional features, ignoring secondary details and enabling the agent to seize the most decision-relevant parts. In the above factorization, the causal part $\mathcal{Z}_c$ has a larger parent set than $\mathcal{Z}_{nc}$. From a causal perspective, this suggests that $\mathcal{Z}_c$ lies closer to the core decision-making path and likely captures higher level and more decision-relevant features. As demonstrated in Fig. 2, s^2 and s^3 have more causal parent

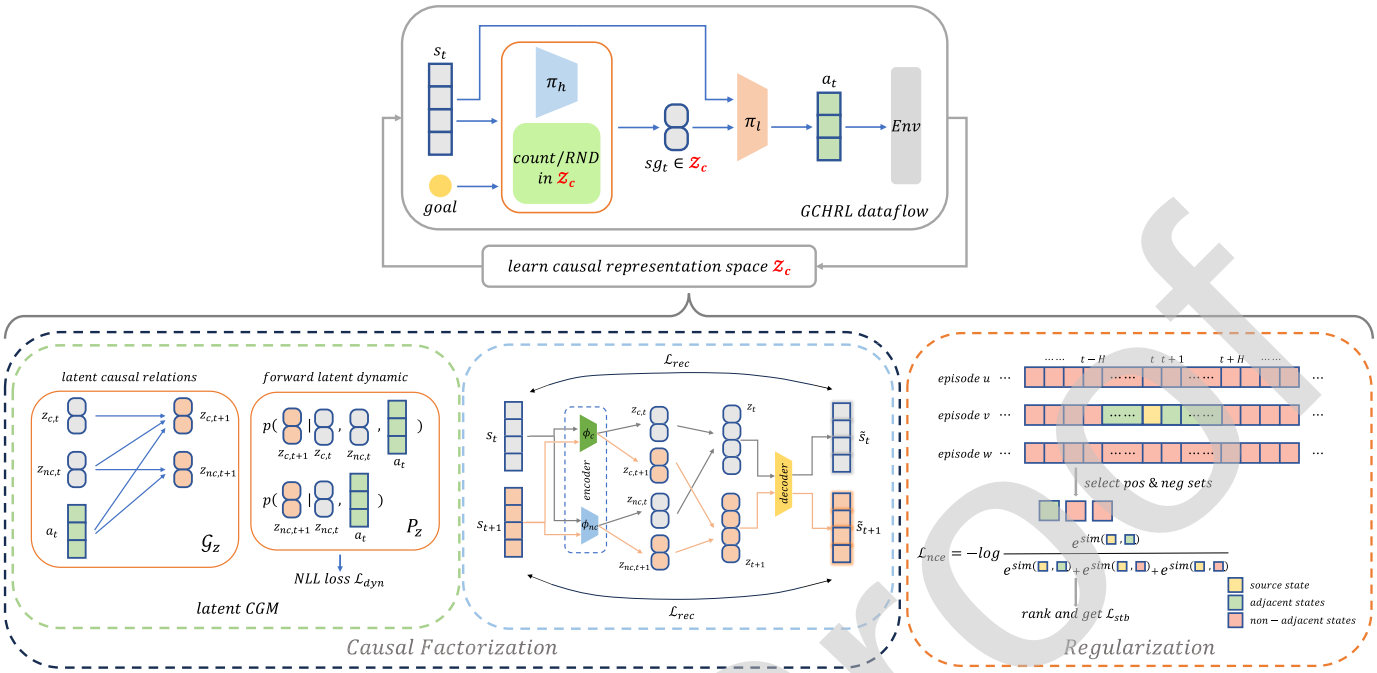


Fig. 1. Framework of CSR. The above part describes the general GCHRL structure, while the below three parts illustrate the components of our representation learning scheme: 1) the latent CGM and the latent causal dynamic transition model, which impose causal consistency constraints on the latent space and extract causal representation from the state space; 2) the reconstruction constraints that supervise the learning process; and 3) the regularization part, which further alleviates collapse and drift in the representation space. In the above workflow, the agent interacts with the environment, collects data, and learns policy; while in the below workflow, it learns the subgoal space. These two parts are carried out iteratively.

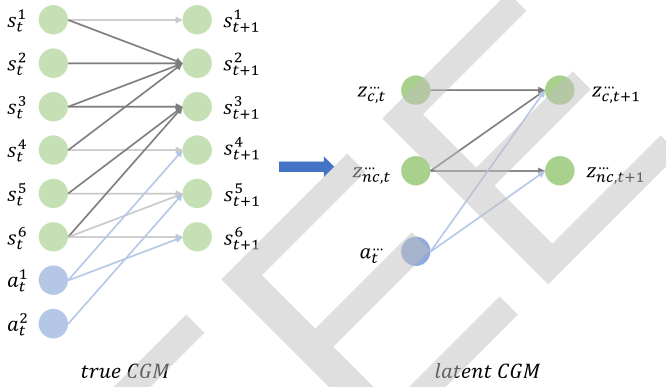


Fig. 2. Demonstration of causal factorization. As s^2 and s^3 have relatively larger causal parent sets, the learned causal representation z_c is likely to contain more information about s^2 and s^3.

nodes than other dimensions, so they are likely to contain more important features. These features are then compressed into $\mathcal{Z}_c$ via representation learning. Thus, we employ $\mathcal{Z}_c$ as the subgoal space.

As $\mathcal{Z}$ is partitioned into two parts, the representation model also contains two parts: $\phi_c : \mathcal{S} \to \mathcal{Z}_c$ and $\phi_{nc} : \mathcal{S} \to \mathcal{Z}_{nc}$. This explicit decoupling of causal and noncausal features differs from prior work. We learn the representation model through two types of constraints: state reconstruction and causal dynamics consistency described in (3). To enable reconstruction of original states from the latent space, we introduce a decoder $d : \mathcal{Z}_c \times \mathcal{Z}_{nc} \to \mathcal{S}$. We denote the latent dynamic models of Z_c and Z_{nc} as $p_c(z_{c,t+1}|z_{c,t}, z_{nc,t}, a_t)$ and $p_{nc}(z_{nc,t+1}|z_{nc,t}, a_t)$, respectively. Both models output parameterized normal distributions over the predicted next-step latent features.

The reconstruction loss $\mathcal{L}_{rec}$ is written as

$$\mathcal{L}_{rec} = \mathbb{E}_{(s_t, s_{t+1})\sim\mathcal{B}} \Big[\big\| s_t - d\left(\phi_c(s_t), \phi_{nc}(s_t)\right) \big\|_2^2 + \big\| s_{t+1} - d\left(\phi_c(s_{t+1}), \phi_{nc}(s_{t+1})\right) \big\|_2^2 \Big] \tag{4}$$

where $\mathcal{B}$ is the buffer. This is essentially an encoder–decoder structure. The causal dynamics loss is calculated by fitting the latent dynamics. We adopt the negative log-likelihood between the predicted distribution and the target value as the causal dynamics loss $\mathcal{L}_{dyn}$

$$\mathcal{L}_{dyn} = \mathbb{E}_{(s_t, s_{t+1})\sim\mathcal{B}} \Big[-\log p\left(\mathbf{sg}\left(z_{c,t+1}\right) \big| \mathbf{sg}\left(z_{c,t}\right), \mathbf{sg}\left(z_{nc,t}\right), a_t\right) - \log p\left(\mathbf{sg}\left(z_{nc,t+1}\right) \big| \mathbf{sg}\left(z_{nc,t}\right), a_t\right) \Big] \tag{5}$$

where $z_{m,n} = \phi_m(s_n), m \in \{c, nc\}, n \in \{t, t+1\}$, and $\mathbf{sg}(\cdot)$ denotes the stop-gradient operation.

The main causal representation loss $\mathcal{L}_{cau}$ is the combination of $\mathcal{L}_{dyn}$ and $\mathcal{L}_{rec}$

$$\mathcal{L}_{cau} = \mathcal{L}_{dyn} + c \cdot \mathcal{L}_{rec} \tag{6}$$

where c is an adjustable coefficient. By optimizing $\mathcal{L}_{cau}$, we can learn a fine subgoal representation model ϕ_c.

We note that this "factorization and reconstruction" paradigm may be more suitable for tasks with explicit transition mechanisms. For high-dimensional image tasks, necessary adaptations may be needed, making it a potential limitation.

B. Regularization on the Representation Space

While the reconstruction constraint mitigates representation collapse to some extent, slight collapses and instability in the representation space can still occur during training, degrading performance. To address this, we propose a regularization technique that consists of two components: the first is designed to further alleviate collapse, and the second aims to mitigate representation drift and enhance stability.

1) Alleviating Representation Collapse With Contrastive Learning: Current methods mainly focus on local temporal adjacency and lack consideration of global relationships in the state space. Our proposed method is partially motivated by this, aiming to emphasize global consistency more.

Assume we have collected M episodes in the buffer $\mathcal{B}$. The ith episode is denoted by $e_i = \{s_{i,0}, \ldots, s_{i,T-1}\}$, where T is the maximum number of timesteps per episode. For two states $s_{i,j}$ and $s_{i,k}$ from the same episode, we consider them nonadjacent if $|j - k| \geq H$. Besides, states from different episodes are also treated as nonadjacent, regardless of their similarity in $\mathcal{S}$. For example, even if $\|s_{u,j} - s_{v,k}\|_2$ is small, we still consider $s_{u,j}$ and $s_{v,k}$ nonadjacent if they come from different episodes. Conversely, two states $s_{u,j}$ and $s_{v,k}$ are regarded as adjacent only if $u = v$ and $|j - k| < H$. Our objective is to bring the representations of adjacent states closer while pushing apart those of nonadjacent states in the latent space. This is realized through contrastive learning, specifically by an InfoNCE-style loss [34]

$$\mathcal{L}_{nce} = -\mathbb{E}_{(s,S_p,S_n)\sim\mathcal{B}} \log \left[\frac{\sum_{s_{p,i}\in S_p} e^{sim(s,s_{p,i})/\tau}}{\sum_{s_{p,i}\in S_p} e^{sim(s,s_{p,i})/\tau} + \sum_{s_{n,j}\in S_n} e^{[sim(s,s_{n,j})+\epsilon]/\tau}} \right] \tag{7}$$

where S_p and S_n denote the sets of positive (i.e., adjacent) and negative (i.e., nonadjacent) samples, τ denotes the temperature coefficient, and $sim(\cdot,\cdot)$ denotes the similarity metric. We define the similarity between two states s_1 and s_2 as the negative L2 distance in the causal latent space, i.e.,

$$sim(s_1, s_2) = -\left\|\phi_c(s_1) - \phi_c(s_2)\right\|_2^2. \tag{8}$$

To better separate nonadjacent states, we add an extra offset $\epsilon > 0$ to the similarity scores of negative samples. This margin ensures $sim(s, s_{n,j})$ does not become excessively large. By optimizing $\mathcal{L}_{nce}$, the similarity of positive samples is maximized while that of negative samples is minimized, enhancing the discriminability of the learned latent space. As negative samples are drawn from the entire interaction dataset rather than being restricted to the same episode, a state with a lower InfoNCE loss indicates that its representation is well separated and globally consistent—optimized with respect to the overall distribution instead of being limited to a single trajectory. This further helps mitigate representation collapse.

2) Stabilizing the Subgoal Space: To stabilize the subgoal space and reduce drift, we propose a regularization loss $\mathcal{L}_{stb}$ that anchors important parts of current causal representations to their prior versions before the network update

$$\mathcal{L}_{stb} = \mathbb{E}_{s\in S_{prio}\cup S_{start}} \left\|\phi_c(s) - \phi_{c,old}(s)\right\|_2^2 \tag{9}$$

where S_{prio} denotes the priority set and S_{start} denotes the set of states near the starting point. Before each network update, we rank states in $\mathcal{B}$ with $\mathcal{L}_{nce}$, and take the top 30% of states with the minimum $\mathcal{L}_{nce}$ to compose S_{prio}. To further prevent spatial drift, we collect states near the starting point, i.e., $\{s_{i,j} | i \in \{1, \ldots, M\}, j \in \{0, \ldots, H-1\}\}$, to compose S_{start}. The above selection strategies for S_{prio} and S_{start} anchor the overall distribution and the starting region, which helps stabilize the entire latent space.

The overall representation loss combines $\mathcal{L}_{cau}$, $\mathcal{L}_{nce}$, and $\mathcal{L}_{stb}$

$$\mathcal{L}_{repr} = \mathcal{L}_{cau} + \alpha \cdot \mathcal{L}_{nce} + \beta \cdot \mathcal{L}_{stb} \tag{10}$$

where α and β are both adjustable coefficients. In practice, $\mathcal{L}_{repr}$ ensures both the quality and stability of the representation space.

C. Hybrid Exploration Strategy in the Subgoal Space

To accelerate exploration, we take advantage of both RND and count-based methods. When applied directly to high-dimensional state spaces, these approaches suffer from limitations: high dimensionality degrades the accuracy of intrinsic rewards in RND and renders space discretization infeasible for count-based exploration. With a well-learned, low-dimensional subgoal space, both RND and count-based methods become much more tractable. By operating in this space, both of them become computationally efficient and effective.

1) Random Network Distillation: The compactness of the subgoal space allows us to use lightweight RND networks. We utilize a predictor network ψ_{pre} and a target network ψ_{tar}, both implemented as three-layer multilayer perceptrons (MLPs). We randomly initialize them and freeze ψ_{tar} throughout training. The intrinsic reward for a state s is defined as

$$r_{int}(s) = min\left\{r_{max}, \left\|\psi_{pre}\left[\phi_c(s)\right] - \psi_{tar}\left[\phi_c(s)\right]\right\|_2^2\right\} \tag{11}$$

where r_{max} is a predefined threshold. At each timestep t, $r_{t,int}$ is computed and added to the original reward. During training, ψ_{pre} is updated periodically by minimizing the mse loss $\mathcal{L}_{rnd}$

$$\mathcal{L}_{rnd} = \mathbb{E}_{s\sim\mathcal{B}}\left\|\psi_{pre}\left[\phi_c(s)\right] - \psi_{tar}\left[\phi_c(s)\right]\right\|_2^2. \tag{12}$$

2) Count-Based Exploration: In addition to RND, state novelty can also be measured by visit counts. We partition the subgoal space into discrete grids and maintain the visit counts for each grid. The visit count of a state s (or a subgoal sg) is represented by $n(\phi_c(s))$ [or $n(sg)$]. When the high-level policy selects a new subgoal at timestep t, it can sample some nearby candidates and choose the one with the lowest visit count, denoted as $sg_{t,v}$. To encourage exploration of less-visited regions, we further extend this subgoal in the direction of $sg_{t,v} - \phi_c(s_t)$, following [7]. The final subgoal sg_t is computed as

$$sg_{t,v} = \underset{sg\in\{sg \,|\, \|sg-\phi_c(s_t)\|_2 < d_g\}}{\arg\min} n(sg)$$
$$sg_t = \phi_c(s_t) + \delta \cdot \left[sg_{t,v} - \phi_c(s_t)\right] \tag{13}$$

Algorithm 1 CSR Algorithm

Input: High-level horizon H, total training episodes $\mathcal{N}$, representation updating frequency f and RND updating frequency f_R.
Initialize: High- and low-level policy π_h and π_l, representation model $\phi = \{\phi_c, \phi_{nc}\}$, visit counter C, select-by-count probability p, RND networks ψ_{pre}, ψ_{tar}, high- and low-level replay buffer $\mathcal{B}_h$ and $\mathcal{B}_l$.
for $i = 1..\mathcal{N}$ **do**
  Sample episode goal g.
  for $t = 0, 1, \ldots, T-1$ **do**
    if $t \equiv 0 (\text{mod } H)$ **then**
      Select sg_t via Eq. (13) in C with probability p; select sg_t by π_h with probability $1-p$.
      Collect high-level transition into $\mathcal{B}_h$.
    else
      Get sg_t by $sg_t = sg_{t-1} + \phi_c(s_{t-1}) - \phi_c(s_t)$.
    end if
    Select a_t with π_l; execute a_t and obtain r_t, s_{t+1} and update C with $\phi_c(s_{t+1})$.
    Compute $r_{t,int}$ with Eq. (11) and add it to r_t.
    Collect low-level transition into $\mathcal{B}_l$.
  end for
  Update π_h, π_l with DRL algorithms.
  if $i \equiv 0 (\text{mod } f)$ **then**
    Update ϕ with Eq. (10).
  end if
  if $i \equiv 0 (\text{mod } f_R)$ **then**
    Update ψ_{pre} with Eq. (12).
  end if
end for
Return: π_h, π_l, ϕ_c.

where d_g is the distance limit and δ is a scaling factor. In practice, the count-based strategy is employed with a probability p to accelerate exploration.

During training, both RND and the count-based strategy are integrated to form a hybrid exploration scheme. The complete training procedure is outlined in Algorithm 1.

V. Experiments

A. Experimental Setup

1) Environments: We conduct experiments on a series of challenging long-horizon control tasks based on the Mujoco [35] simulator, including various locomotion and manipulation tasks. Specifically, the experiments are conducted on the following seven environments.

1) *AntMaze (U-Shape):* A simulated ant robot starts at $(0, 0)$. The edge length of the maze is 12. In the training phase, the goal is randomly sampled within the maze (this setting is the same for all ant tasks), while in the evaluation phase, the goal is fixed to the farthest point $(0, 8)$. The maximum timestep for each episode is set to 500 for training and evaluation.
2) *AntMaze (S-Shape):* The starting point is $(0, 0)$. The maze length is 20. In the evaluation phase, the goal is the farthest point $(16, 16)$. The maximum episode timestep is 1000.
3) *AntMaze (W-Shape):* The starting point is $(0, 0)$. The length of the maze is 20. In the evaluation phase, the goal is the farthest point $(0, 8)$. The maximum episode timestep is 1000.
4) *AntFourRooms:* The ant starts at the bottom left room and aims to reach the top right room. The initial xy position is $(0, 0)$. The edge length of the maze is 18. In the evaluation phase, the goal is fixed to $(14, 14)$. The maximum episode timestep is 1000.
5) *AntPush:* The ant starts at the bottom. To reach the goal at the top, it must push the obstacle to the right first. The initial xy position is $(0, 0)$. The edge length of the environment is 12. In the evaluation phase, the goal is fixed to $(0, 8)$ above the obstacle. The maximum episode timestep is 500.
6) *HalfCheetahHurdle:* A HalfCheetah robot is required to jump over a hurdle of height 0.35 and then reach the goal. The initial xy position is $(0, 0)$. The x-coordinate of the hurdle is set to 2, and the height of the hurdle is 0.35. The goal is fixed to $(4, 0.2)$ for both training and evaluation. The maximum episode timestep is 1000.
7) *3-D-Reacher:* A 7-DOF robot arm is situated in front of a table. Its gripper is expected to reach specified 3-D goal positions. The robot arm starts at a fixed position; the initial angles and angular velocities of each joint are all 0. For both training and evaluation, the goal is randomly sampled. The maximum episode timestep is 100.

Note that the rewards are sparse and binary for all these environments. For Ant tasks, when the L2 distance between the xy coordinates of the ant robot and the desired goal is less than 1.5, the ant robot achieves a "success" and gets a reward of 1; otherwise, the reward is always 0. For the HalfCheetahHurdle task, when the L2 distance between the xy coordinates of the cheetah robot and the desired goal is less than 0.5, the cheetah robot achieves a "success" and gets a reward of 1; otherwise, the reward is 0. For the Reacher task, when the L2 distance between the xyz coordinates of the gripper and the desired 3-D goal position is less than 0.25, the robot achieves a "success" and gets a reward of 1; otherwise, the reward is 0.

The above environments are displayed in Fig. 3.

2) Baselines: CSR is compared with the following baseline methods.

1) *LESSON [10]:* A GCHRL method that learns subgoal representations based on slow dynamics.
2) *HESS [7]:* An extension of LESSON that adds stabilization regularization and employs an active exploration scheme based on novelty and reachability of latent subgoals.
3) *HLPS [24]:* A GCHRL method that learns the subgoal space from a probabilistic perspective.
4) *HRAC [21]:* A GCHRL method that uses predefined xy space as the subgoal space directly and restricts subgoal selection with temporal adjacency.

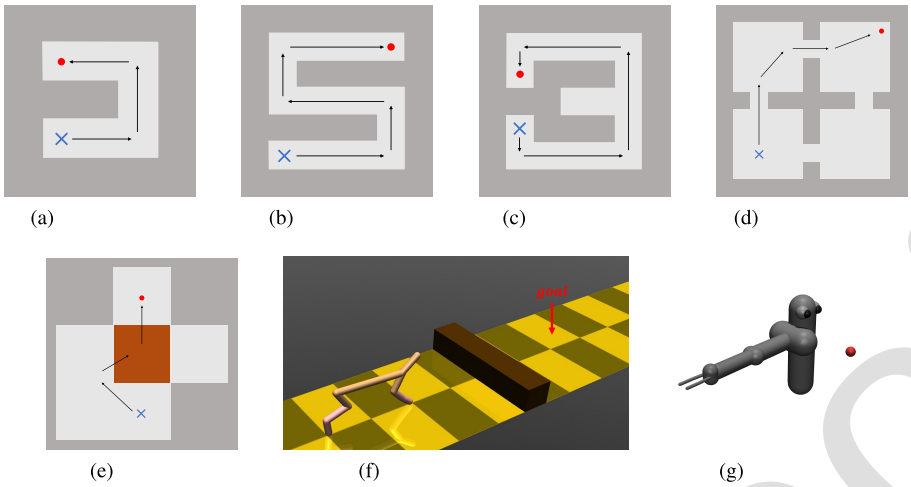


Fig. 3. Environments used in our experiments. (a) AntMaze(U). (b) AntMaze(S). (c) AntMaze(W). (d) AntFourRoom. (e) AntPush. (f) HalfCheetahHurdle. (g) Reacher.

TABLE I
COMPARISON AMONG CSR AND BASELINES

Method	Representation	Causal modeling	Exploration strategy
CSR	✓	✓	rnd+count
HESS	✓	✗	novelty+success
LESSON	✓	✗	random noise
HLPS	✓	✗	random noise
HRAC	✗	✗	random noise
SAC	✗	✗	random noise

5) *SAC [36]:* A representative non-HRL algorithm that refines reinforcement learning with policy entropy, enhancing exploration.

Table I compares CSR and the above methods along several dimensions (whether to use representation learning, whether to use causal modeling, and exploration strategies).

B. Comparative Analysis

For each algorithm and environment, we evaluate with ten episodes in the corresponding test environment. We compute the success rate after every 50 000 timesteps. The average success rate over five runs and the corresponding 95% confidence interval are reported as the performance metric.

From Fig. 4, we can see that CSR outperforms baseline methods on most tasks, especially on harder ones such as the S- and W-shaped AntMaze, demonstrating its high learning efficiency. The performance of CSR relies on its stable and well-learned subgoal space, as well as the exploration strategies applied to it. Taking advantage of the underlying causal relations among state variables, CSR composes the subgoal space using the most causally compact part, which is more reliable and explainable compared to other approaches.

In comparison, LESSON and HESS learn subgoal spaces mainly with local adjacency information, so global information may be ignored, degrading the representation quality and the overall performance. HLPS leverages a Gaussian process to model subgoal spaces, improving the compatibility of new states. However, it exhibits relatively low exploration capability, which hinders effective updates of posterior distributions. On the other hand, too frequent updates may impact stability, which also adversely affects policy learning. HRAC utilizes a predefined subgoal space, which is inherently stable and well-structured. However, it leverages adjacency information to restrict subgoal selection, which significantly enhances exploitation but also overemphasizes it. Thus, exploration is limited, making it less efficient. In addition, *xy* subgoal space may not be very suitable for some tasks, which limits its further applications. As a non-HRL approach, SAC relatively lacks the capacity for tackling complex and sparse-reward tasks. Although it encourages exploration with policy entropy, the large state spaces of these tasks are difficult for efficient exploration. Therefore, it cannot learn effective policies within several million timesteps.

C. Ablation Study

To verify the effectiveness of various components, we conduct ablation studies on them.

1) Ablation Study on Exploration Strategies: To verify the effectiveness of exploration methods applied in the subgoal space, we conduct ablation studies with different exploration strategies on the AntFourRooms task. The results presented in Fig. 5(a) indicate that both RND and the count-based method contribute to the overall performance.

2) Ablation Studies on Regularizations: We conduct ablation studies on the U-shaped AntMaze task to examine the impact of the $\mathcal{L}_{nce}$ coefficient α [see Fig. 5(b)] and the $\mathcal{L}_{stb}$ coefficient β [see Fig. 5(c)].

The results in Fig. 5(b) suggest that neither too large nor too small α values are optimal. Too large α imposes excessive regularization on the subgoal space and may reduce uniformity, while too small α provides insufficient regularization and cannot mitigate representation collapse well.

From Fig. 5(c), we can see that the stable loss also significantly influences overall performance. Large β values hinder the updates of ϕ, which is harmful to training. On the other

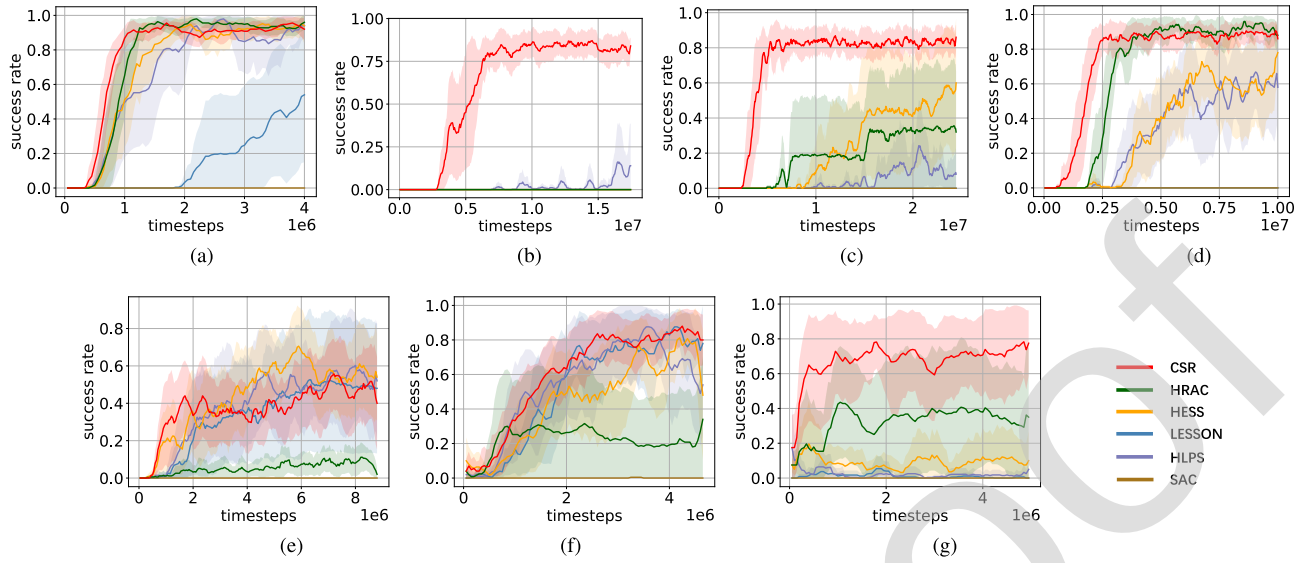


Fig. 4. Learning curves on a series of tasks. All curves are smoothed equally for visual clarity. (a) AntMaze (U-shape). (b) AntMaze (S-shape). (c) AntMaze (W-shape). (d) AntFourRooms. (e) AntPush. (f) HalfCheetahHurdle. (g) Reacher.

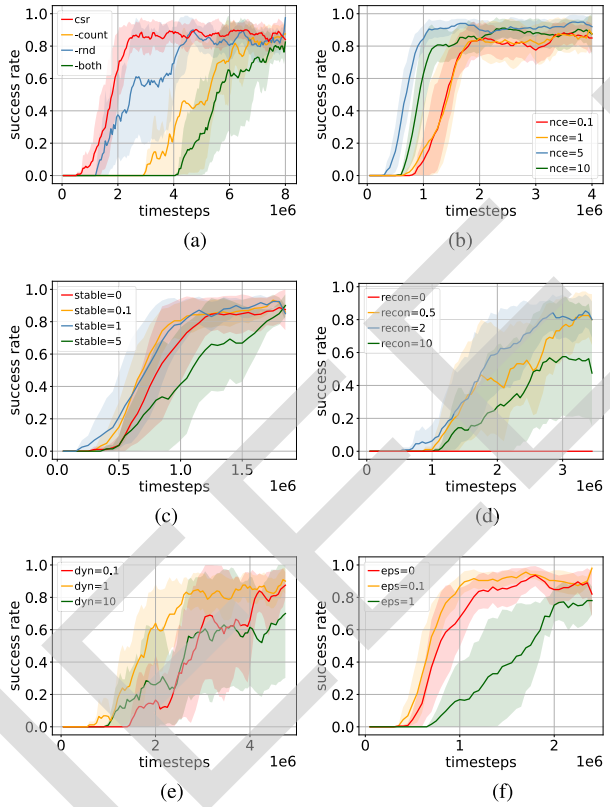


Fig. 5. Ablation studies on various components of CSR. (a) Exploration. (b) InfoNCE coefficients. (c) Stable coefficients. (d) Reconstruction. (e) Causal dynamics. (f) Values of ϵ.

hand, the learning efficiency will also be influenced without the stable loss.

3) Ablation Studies on Causal Losses: We conduct experiments on the AntFourRooms task to reveal the effectiveness of causal representation learning. Fig. 5(d) illustrates the ablation results under different coefficients c of the reconstruction loss $\mathcal{L}_{rec}$. When $c = 0$, the agent cannot learn effective policies, and the success rate is always 0. On the contrary, nonzero values of c ensure effective learning.

Fig. 5(e) shows the results under different coefficients of the causal dynamic loss. We evaluate with coefficients 0.1, 1, and 10. The results indicate that coefficients that are either too large or too small are not optimal. Based on this, we set the coefficient to 1 as the default choice.

4) Ablation Study on InfoNCE Offsets: We examine different values of ϵ on the U-shaped AntMaze. The results shown in Fig. 5(f) indicate that it is proper to set a relatively small ϵ to assist the regularization. Too large ϵ may overly separate representations, leading to distortion of the subgoal space.

TABLE II
HYPER-PARAMETERS ACROSS ALL ENVIRONMENTS

Hyperparameters	Values
Causal representation dimension	2
High-level action interval H	20
Learning rate for both levels of actor networks	0.0001
Learning rate for both levels of critic networks	0.001
Discount factor for both levels γ	0.99
Learning rate for representation models	0.0001
Learning rate for dynamic models	0.0002
Coefficient of $\mathcal{L}_{dyn}$	1
Coefficient of $\mathcal{L}_{rec}$ c	2
Offset added to $\mathcal{L}_{nce}$ ϵ	0.1
Coefficient of $\mathcal{L}_{stb}$ β	0.1
Scaling factor δ in count-based subgoal selection	3
Distance limit d_g	10
Probability of using count-based selection p	0.3
Learning rate of RND	0.0002
Threshold of intrinsic reward r_{max}	0.8
Grid size for count-based method	1
Representation updating frequency f	100
RND updating frequency f_R	50

D. Analysis on Representation Spaces

For comprehensive validation, we visualize subgoal spaces learned by CSR, HESS, and HLPS. For each method, we

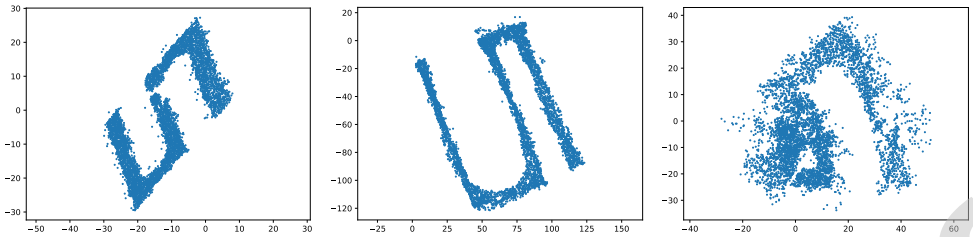


Fig. 6. Learned subgoal spaces of different methods. (a) CSR. (b) HESS. (c) HLPS.

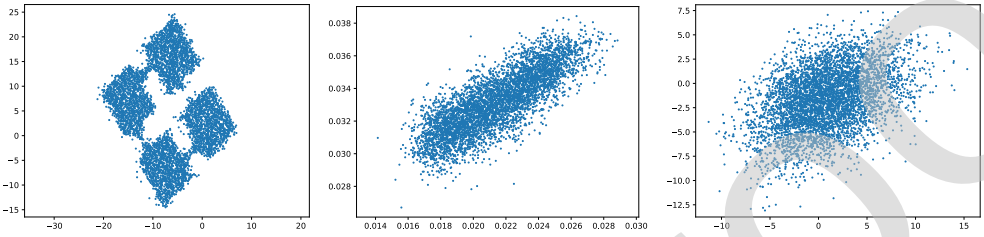


Fig. 7. Comparison of subgoal spaces with and without the causal loss $\mathcal{L}_{cau}$. (a) CSR. (b) CSR without $\mathcal{L}_{rec}$. (c) CSR without $\mathcal{L}_{dyn}$.

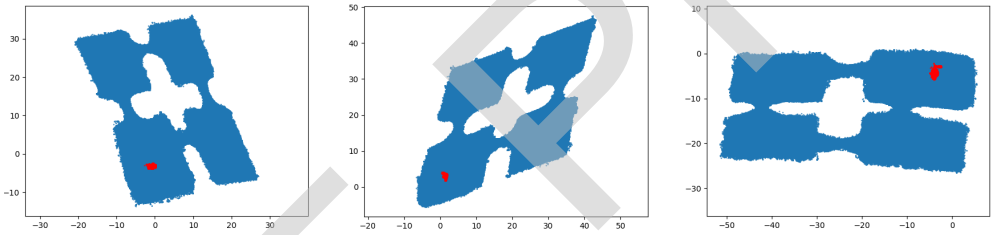


Fig. 8. Comparison of learned subgoal spaces under different random seeds. It can be seen that they are similar under affine transformation, proving CSR's robustness.

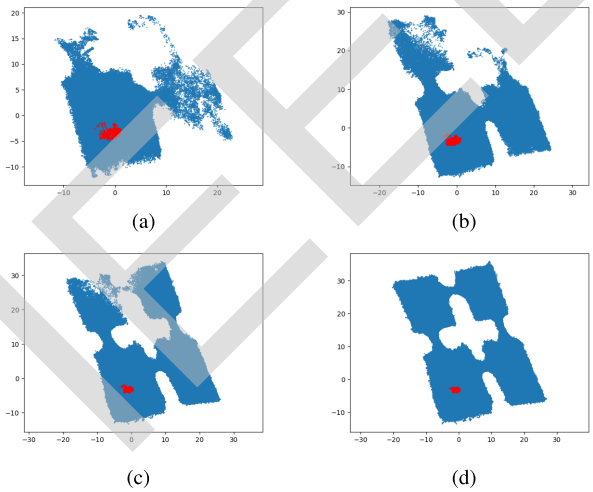


Fig. 9. Dynamics of the learned subgoal space in the AntFourRooms task. It can be seen that the subgoal space is generally stable. No drastic drifts are exhibited. (a) 500 episodes. (b) 1000 episodes. (c) 2000 episodes. (d) 3000 episodes.

randomly sample an equal number of states in the S-shaped AntMaze environment and map them to the subgoal space using the learned representation model. As illustrated in Fig. 6,

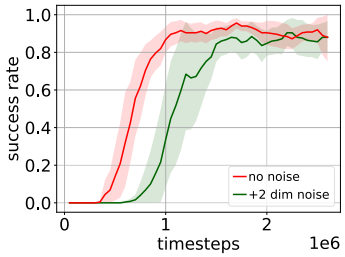
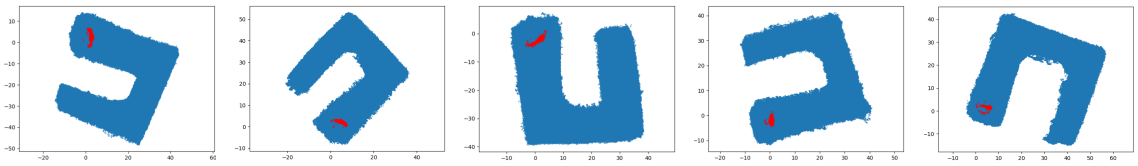


Fig. 10. Comparison between original curve and curve with distractors. The performance is affected by noise, but not severely.

the subgoal space of CSR has strong uniformity and little collapse. In contrast, the subgoal space of HESS is inadequate in uniformity, posing a risk of representation collapse. The subgoal space of HLPS lacks regularity and suffers from relatively severe collapse, which degrades its performance.

In addition, we visualize the subgoal space learned without the reconstruction loss $\mathcal{L}_{rec}$ or the dynamic loss $\mathcal{L}_{dyn}$ in the AntFourRooms environment for comparison with that learned by original CSR. The subgoal space shown in Fig. 7(b) and (c) both exhibits very poor performance compared to Fig. 7(a). It can be seen that the agent cannot learn meaningful representations without any component of $\mathcal{L}_{cau}$. Without the



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

Fig. 11. Representation spaces learned under five different seeds. It can be seen that noise is almost not included in the learned spaces, which proves CSR's effectiveness in filtering task-irrelevant information.

TABLE III
HYPER-PARAMETERS THAT DIFFER ACROSS THE ENVIRONMENTS

Hyperparameters	AntMaze-U	AntMaze-S	AntMaze-W	AntFourRooms	AntPush	HalfCheetahHurdle	Reacher
Non-causal representation dimension	10	10	10	10	10	8	8
Coefficient of $\mathcal{L}_{nce}$ α	5	0.1	0.1	0.1	0.1	1	0.1

dynamic loss, the representation model cannot decide what to preserve and what to discard. On another aspect, the representation model will lose important information for supervised learning without reconstruction. These further demonstrate the importance of causal representation learning.

To evaluate CSR's representation learning quality across different runs, we visualize subgoal spaces obtained under several random seeds in the AntFourRooms task, shown in Fig. 8. The red regions indicate neighborhoods of the starting position. The resulting spaces are observed to be similar up to affine transformation, demonstrating CSR's insensitivity to randomness. Since CSR adopts relative subgoals, absolute positions do not affect policy learning.

Finally, to investigate the variation of the representation space over time, we visualize the space after several training episodes, as shown in Fig. 9. It can be seen that the space is generally stable and exhibits no drastic drifts.

E. Robustness to Task-Irrelevant Distractors

To demonstrate CSR's ability to filter out task-irrelevant distractors, we add two dimensions of random Gaussian noise $[\mathcal{N}(0, 2^2)]$ to the state space of the U-shaped AntMaze environment and evaluate CSR under five random seeds. The learning curve and learned representation spaces are shown in Figs. 10 and 11, respectively. The results indicate that overall performance is not severely influenced by the noise, and CSR remains capable of extracting a valid subgoal space from the noisy state space.

VI. CONCLUSION AND DISCUSSION

In this article, we propose CSR, an effective approach for learning subgoal representation spaces and enhancing exploration in GCHRL. Leveraging causally factorized latent dynamics, CSR extracts more causally important and decision-relevant features from the state space, enabling more efficient policy learning. In addition, CSR further enhances the overall performance through regularization and exploration strategies.

It is important to note that CSR is fundamentally a subgoal representation learning approach, so it can be easily incorporated into other GCHRL algorithms to gain further enhancements. Moreover, compared to other self-supervised schemes, CSR benefits from supervised signals, leading to lower time consumption for representation learning.

CSR also has some limitations. For instance, we have mentioned that this paradigm may not be suitable for tasks with image observations. The high dimensionality of image observations may exceed the reconstruction capabilities of simple networks (such as MLPs), and there may not exist explicit causal relations among raw pixels. One potential improvement is to incorporate more advanced architectures, such as CNN [37] and ViT [38], to extract high-level features from image observations. However, this may increase computational overhead. Therefore, one interesting future work is to adapt CSR to these tasks.

APPENDIX: IMPLEMENTATION DETAILS

A. Network Structure

1) Actor and Critic Networks: Actor and critic for both high- and low-level policies are implemented as MLPs. Each MLP has two hidden layers of dimension 300 with ReLU activations. For actors of both levels, the outputs are scaled to $[-1, 1]$ by the Tanh function and then scaled to the range of action spaces using linear transformations.

2) Representation Networks: ϕ contains two components: the causal encoder ϕ_c and the noncausal one ϕ_{nc}, both of which are MLPs with two hidden layers of dimension 128. ReLU activations are applied in both models.

3) Dynamics Networks: Corresponding to the encoder structure, the latent dynamics network is also two-part: the causal dynamics network and the noncausal dynamics network. We implement each of them with a simple multihead self-attention block. The number of heads is 4, and the dimension of embedding is 32 for each head.

Specifically, for each dimension of latent features or primitive actions, it is mapped to the embedding space by an MLP with one hidden layer of dimension 32. Then, these embeddings are processed by the self-attention block. After attention, we utilize two single-layer MLPs for each embedding to compress it to predictions of next-step μ and $\log \sigma$, both of

which are scalars. Dimensions corresponding to latent features to be predicted are retained, while others are discarded.

4) Decoder Network: The decoder network maps latent features to the original state space $\mathcal{S}$, i.e., $d : \mathcal{Z}_c \times \mathcal{Z}_{nc} \to \mathcal{S}$. It is implemented as an MLP with two hidden layers of dimension 128 and ReLU activations.

5) RND Networks: RND consists of two networks: a predictor ψ_{pre} and a target network ψ_{tar}. We implement both of them with MLPs with two hidden layers of dimension 128. ReLU activations are employed. Parameters of ψ_{tar} are frozen and parameters of ψ_{pre} are learnable.

B. Hyperparameters

The choices of main hyperparameters in our experiments are displayed in Tables II and III. In Table III, for the convenience of layout, we use "AntMaze-U," "AntMaze-S," and "AntMaze-W" to denote the U-shaped AntMaze, the S-shaped AntMaze, and the W-shaped AntMaze, respectively.

REFERENCES

[1] S. Pateria, B. Subagdja, A.-H. Tan, and C. Quek, "Hierarchical reinforcement learning: A comprehensive survey," *ACM Comput. Surv.*, vol. 54, no. 5, pp. 1–35, Jun. 2022.

[2] P.-L. Bacon, J. Harb, and D. Precup, "The option-critic architecture," in *Proc. AAAI Conf. Artif. Intell.*, 2017, vol. 31, no. 1, pp. 1726–1734.

[3] A. Levy, G. Konidaris, R. Platt, and K. Saenko, "Learning multi-level hierarchies with hindsight," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–16.

[4] Z. Yang, K. Merrick, L. Jin, and H. A. Abbass, "Hierarchical deep reinforcement learning for continuous action control," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5174–5184, Nov. 2018.

[5] N. Dilokthanakul, C. Kaplanis, N. Pawlowski, and M. Shanahan, "Feature control as intrinsic motivation for hierarchical reinforcement learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 11, pp. 3409–3418, Nov. 2019.

[6] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 3826–3839, Sep. 2020.

[7] S. Li, J. Zhang, J. Wang, Y. Yu, and C. Zhang, "Active hierarchical exploration with stable subgoal representation learning," in *Proc. Int. Conf. Learn. Represent.*, vol. 10, 2022, pp. 1–18.

[8] Y. Luo, F. Sun, T. Ji, and X. Zhan, "Bidirectional-reachable hierarchical reinforcement learning with mutually responsive policies," in *Proc. Reinf. Learn. Conf.*, 2024. AQ:7

[9] H. Wang, Z. Tang, Y. Sun, F. Wang, S. Zhang, and Y. Chen, "Guided cooperation in hierarchical reinforcement learning via model-based rollout," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 5, pp. 8455–8469, May 2025.

[10] S. Li, L. Zheng, J. Wang, and C. Zhang, "Learning subgoal representations with slow dynamics," in *Proc. Int. Conf. Learn. Represent.*, 2021, pp. 1–18.

[11] Z. Deng, J. Jiang, G. Long, and C. Zhang, "Causal reinforcement learning: A survey," *IEEE Trans. Mach. Learn. Res.*, vol. 2023, no. 11, pp. 1–52, Nov. 2023.

[12] Z. Deng, H. Tian, X. Zheng, and D. D. Zeng, "Deep causal learning: Representation, discovery and inference," *ACM Comput. Surv.*, vol. 58, no. 2, pp. 1–36, Jan. 2026.

[13] Z. Wang, X. Xiao, Z. Xu, Y. Zhu, and P. Stone, "Causal dynamics learning for task-independent state abstraction," in *Proc. Int. Conf. Mach. Learn.*, vol. 162, 2022, pp. 23151–23180.

[14] Z. Yu, J. Ruan, and D. Xing, "Explainable reinforcement learning via a causal world model," in *Proc. 32nd Int. Joint Conf. Artif. Intell.*, Aug. 2023, pp. 4540–4548.

[15] W. Ding, H. Lin, B. Li, and D. Zhao, "Generalizing goal-conditioned reinforcement learning with variational causal reasoning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 26532–26548.

[16] A. Zhang, R. McAllister, R. Calandra, Y. Gal, and S. Levine, "Learning invariant representations for reinforcement learning without reconstruction," in *Proc. Int. Conf. Learn. Represent.*, vol. 9, 2021, pp. 1–16.

[17] Y. Burda, H. Edwards, A. Storkey, and O. Klimov, "Exploration by random network distillation," in *Proc. 7th Int. Conf. Learn. Represent.*, 2019, pp. 1–17.

[18] O. Nachum, S. S. Gu, H. Lee, and S. Levine, "Data-efficient hierarchical reinforcement learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–11.

[19] M. Andrychowicz et al., "Hindsight experience replay," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[20] S. Pitis, H. Chan, S. Zhao, B. Stadie, and J. Ba, "Maximum entropy gain exploration for long horizon multi-goal reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 7750–7761.

[21] T. Zhang, S. Guo, T. Tan, X. Hu, and F. Chen, "Generating adjacency-constrained subgoals in hierarchical reinforcement learning," in *Proc. NIPS*, 2020, pp. 21579–21590.

[22] S. Pateria, B. Subagdja, A.-H. Tan, and C. Quek, "End-to-end hierarchical reinforcement learning with integrated subgoal discovery," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 7778–7790, Dec. 2022.

[23] J. Kim, Y. Seo, S. Ahn, K. Son, and J. Shin, "Imitating graph-based planning with goal-conditioned policies," in *Proc. Int. Conf. Learn. Represent.*, vol. 11, 2023, pp. 1–19.

[24] V. H. Wang, T. Wang, W. Yang, J.-K. Kinen, and J. Pajarinen, "Probabilistic subgoal representations for hierarchical reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, vol. 41, 2024, pp. 51755–51770.

[25] Z. Yu, J. Ruan, and D. Xing, "Learning causal dynamics models in object-oriented environments," in *Proc. Int. Conf. Mach. Learn.*, vol. 41, 2024, pp. 57597–57638.

[26] A. Zhang et al., "Invariant causal prediction for block MDPs," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 11214–11224.

[27] C. Boutilier, R. Dearden, and M. Goldszmidt, "Stochastic dynamic programming with factored representations," *Artif. Intell.*, vol. 121, nos. 1–2, pp. 49–107, Aug. 2000.

[28] T. Dean and K. Kanazawa, "A model for reasoning about persistence and causation," *Comput. Intell.*, vol. 5, no. 2, pp. 142–150, Feb. 1989.

[29] K. Chalupka, P. Perona, and F. Eberhardt, "Fast conditional independence test for vector variables with large sample sizes," *Stat*, vol. 1050, p. 8, Jan. 2018.

[30] J. Guo, M. Gong, and D. Tao, "A relational intervention approach for unsupervised dynamics generalization in model-based reinforcement learning," in *Proc. Int. Conf. Learn. Represent.*, vol. 10, 2022, pp. 1–24.

[31] H. Cao et al., "Rethinking state disentanglement in causal reinforcement learning," *Comput. Res. Rep.*, 2024. AQ:8

[32] G. Ostrovski, M. G. Bellemare, A. Van Den Oord, and R. Munos, "Count-based exploration with neural density models," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2721–2730.

[33] A. Jonsson and A. Barto, "A causal approach to hierarchical decomposition of factored MDPs," in *Proc. 22nd Int. Conf. Mach. Learn.*, 2005, pp. 401–408.

[34] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.

[35] E. Todorov, T. Erez, and Y. Tassa, "MuJoCo: A physics engine for model-based control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 5026–5033.

[36] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1861–1870.

[37] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[38] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. 9th Int. Conf. Learn. Represent.*, Oct. 2020, pp. 1–22.