CAUSAL RL AGENTS FOR OUT-OF-DISTRIBUTION GENERALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Out-of-distribution (OOD) generalization is critical for applying reinforcement learning algorithms to real-world applications. To address the OOD problem, recent works focus on learning an OOD adaptation policy by capturing the causal factors affecting the environmental dynamics. However, these works recover the causal factors with only an entangled or binary form, resulting in a limited generalization of the policy that requires extra data from the testing environments. To break this limitation, we propose Generalizable Causal Reinforcement Learning (GCRL) to learn a disentangled representation of causal factors, on the basis of which we learn a policy that achieves the OOD generalization without extra training. For capturing the causal factors, GCRL deploys a variant of β-VAE structure with a two-stage constraint to ensure that all factors can be disentangled. Then, to achieve the OOD generalization through causal factors, we adopt an additional network to establish the dependence of actions on the learned representation. Theoretically, we prove that while the optimal policy can be found in training environments, the established dependence can recover the causal relationship between causal factors and actions. Experimental results show that GCRL achieves the OOD generalization on eight benchmarks from Causal World and Mujoco. Moreover, the policy learned by our model is more explainable, which can be controlled to generate semantic actions by intervening in the representation of causal factors.

1 INTRODUCTION

Reinforcement Learning (RL) provides a powerful framework that can train an agent to take proper sequential actions based on the states (Sutton & Barto, 1998; Silver et al., 2017; Vinyals et al., 2019). To apply the RL model in reality, we expect to generalize the agents on factors of variation that are not in the training distribution, which is defined as the OOD problem (Shen et al., 2021; Cui & Athey, 2022; Krueger et al., 2021). An example of the OOD setting is shown in Figure 1. Each instance of an environment is sampled from a distribution, where the testing (red dots) and the training environments (blue dots) belong to different distributions. Most success stories focus on the OOD adaptation that accesses extra data from the testing environments (Sontakke et al., 2021; Zintgraf et al., 2019; Zhou et al., 2021; Mendonca et al., 2020). However, the strict OOD generalization could not interact with testing environments because the testing distribution is commonly unknown in real-world applications (Kirk et al., 2021). Thus, to develop OOD generalization in RL, an effective generalizable policy that only accesses data from training environments is urgently needed.

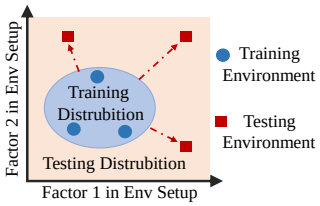


Figure 1: The OOD setting in RL, where causal factors affect the environmental dynamics.

One of the solutions for the OOD problem is to discover the factors of variation that affect the environmental dynamics, from which a generalizable policy can be learned (Killian et al., 2017; Doshi-Velez & Konidaris, 2016). Recent works (Zintgraf et al., 2019; Perez et al., 2020; Yao et al., 2018) propose to discover the factors of variation by modeling them as hidden parameters in the transition function of each environment. Unfortunately, the factors of variation encoded by hidden parameters are prone to entangle and uninterpretable due to the cumulative nature of the influence from the variation. To address this issue, Sontakke et al. (2021) propose Causal Curiosity (CC) to

learn a disentangled representation and formally define these factors of variation as causal factors that are related to environmental dynamics. By learning how the causal factors affect the policy required to achieve the task, agents can generalize to previously unencountered environments. However, CC only considers a binary representation, leading to the policy with a limited generalization that still requires extra data from the testing environments. To go a step further in this direction, we aim to deploy the policy without any extra data by recovering the more precise disentangled representations of causal factors.

The unsupervised learning of disentangled representations without inductive biases has been proven theoretically infeasible (Locatello et al., 2019). Ever since then, substantial efforts (Locatello et al., 2020; Shu et al., 2020) have focused on the weakly-supervised method, which requires the values of some latent factors are shared among paired inputs, such as two adjacent frames in the environment. To adapt the above approaches to RL scenarios, we need an accessible signal that can disentangle the causal factors from other features in the states. Fortunately, recent studies on environmental dynamics find that causal factors are specific to environmental change (Sontakke et al., 2021). Consider the example of a robot lifting a cube, the coordinate of the cube might be changed by robotic arms, but the mass of the cube should only depend on the environmental setting. Therefore, the causal factor is commonly defined as a constant in a single environment but may differ across environments (Zhang et al., 2021b; Huang et al., 2022), making it possible to be disentangled from other features.

Based on the natural characteristics of causal factors, we introduce Generalizable Causal Reinforcement Learning (GCRL), a novel method to learn a disentangled representation, which is used to build a policy that achieves the OOD generalization. GCRL operates in multiple training environments and does not assume access to extra data from the testing environments. For capturing the causal factors, we employ a variant of β-VAE (Higgins et al., 2017) with a two-stage constraint to ensure that all factors can be disentangled. Then, to achieve the OOD generalization through causal factors, we adopt an additional network to establish the dependence of actions on the learned representation. Our contributions are as follows:

- We propose GCRL, a novel weakly-supervised reinforcement learning algorithm, to discover the causal factors that affect the environmental dynamics, and a policy that achieves the OOD generalization without extra data from the testing environments.
- We theoretically prove that RL agents learned with representations of causal factors are general towards unseen environments. The established dependence can recover the causal relationship between causal factors and actions as the policy approaches optimality.
- We empirically demonstrate that GCRL is capable of outperforming the state-of-the-art representation of causal factors. In addition, our RL agent is more explainable and can generate sequential semantic actions by intervening in representations.

2 Problem Formulation

Following (Sontakke et al., 2021), we consider a set of environments that can be accessed by the agent in the training stage and another set of environments only accessed at the testing stage. Each instantiation of an environment has a unique setting for causal factors that are sampled from a different (wider) distribution at the testing stage.

Causal MDPs In Causal MDPs (Sontakke et al., 2021), an RL task can be modeled as a tuple $G =< S, A, P, R, \gamma >$. The state $s \in S$ can be divided into two portions - the controllable state s^c and uncontrollable state s^u. The uncontrollable portion of the state s^u consists of the causal factors of the environment, and the controllable state s^c can be influenced by policy. In each environment step t, the policy $\pi(\cdot|s_t)$ chooses an action $a_t \in A$ depending on the current state $s_t \in S$. Based on the action a_t and the transition function $P(s^c_{t+1}, s^u_{t+1}|s^c_t, s^u_t, a_t)$, the environment transfers to the next state (s^c_{t+1}, s^u_{t+1}) and returns a reward r_t. The aim of the agent is to maximize the cumulative reward $\mathbb{E}_\pi[\sum_{t=1}^{T} \gamma^{t-1} r_t]$, where γ denotes the discount factor. In the same environment, s^u remains constant, i.e., $\forall t < T, s^u_t = s^u_{t+1}$. Inspired by (Bica et al., 2021), we want to learn a policy $\pi(\cdot|s^c_t, s^u_t)$ that depends on causal factors recovered from the state.

Disentangled representations for Causal Factors An environment could be regarded as a generative model, such as the state s is generated from the distribution $p(s|z)$, where z is drawn from a set

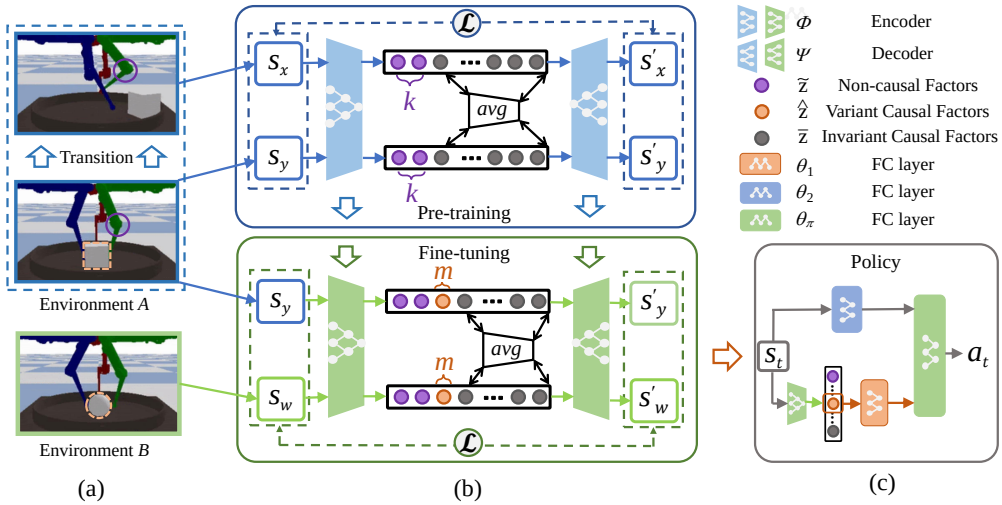


Figure 2: The GCRL framework. (a) Environment setup where objects in scene A and scene B are different. (b) The weakly-supervised generative model for disentangled representations of causal factors. (c) The policy network with representations of the variant causal factors $\hat{z}$.

of independent ground-truth factors of variation $p(z) = \prod_i p(z_i)$ (Locatello et al., 2020). Based on the Causal MDPs, $z = [\tilde{z}, \hat{z}, \overline{z}] \in \mathbb{Z}$, where $\mathbb{Z}$ is the set of factors generating s, $\tilde{z}$ corresponds to s^c, and the remaining parts correspond to s^u. In a single environment, $\hat{z}$ and $\overline{z}$ are equivalent because all causal factors remain constant. For example, while a robotic arm pushes an item, the coordinates of the item ($z_i \in \tilde{z}$) may be changed, but causal factors will remain constant, such as gravity. However, some (but not all) causal factors may vary with the settings, such as some properties (size, mass) of the items being different in multiple environments. Thus, we denote $\hat{z}$ as the variant part, and $\overline{z}$ refers to the invariant part (gravity, etc.) in multiple environments. The goal of disentangled representation learning is to learn a function $f(s)$ mapping the state s to a vector that contains information about causal factors, with each coordinate containing information about only one factor.

Weakly-supervised generative model for RL states Inspired by Shu et al. (2020), we utilize paired states to learn disentangled representations of causal factors, where some factors of variation have the same value. This can be modeled as sampling two states from the trajectories of the environments, as illustrated in Figure 2. Formally, the generative model is given by

$$p(z') = \prod_{i=1}^{d} p(z_i), \quad p(\tilde{z}) = \prod_{i=1}^{k} p(\tilde{z}_i), \quad p(\hat{z}) = \prod_{i=1}^{m} p(\hat{z}_i), \quad p(\overline{z}) = \prod_{i=1}^{d-(k+m)} p(\overline{z}_i), \tag{1}$$

$$s_x = g^*(z'), \quad s_y = g^*([\tilde{z}, \hat{z}, \overline{z}]^T), \tag{2}$$

where z' and $[\tilde{z}, \hat{z}, \overline{z}] \in \mathbb{Z}$, d is the dimension of the encoded vector, k is the number of factors that correspond to s^c in two states, and m is the number of variant causal factors. The generative mechanism is modeled using a function $g^* : z \to s$, which maps the factors to the reconstructed states. To make the relation between s_x and s_y explicit, we add the shared constraint $z'_{k+1:d} = [\hat{z}, \overline{z}]$ in a single environment. As s_x and s_y are sampled from different environments, the constraints on $\hat{z}$ and $\overline{z}$ are further redefined as $z'_{k+1:k+m} \neq \hat{z}$ and $z'_{k+m:d} = \overline{z}$. We aim to infer the disentangled representation $\hat{z}$ to approximate a subset of s^u that varies across environments.

3 Generalizable Causal Reinforcement Learning

The goal of Generalizable Causal Reinforcement Learning (GCRL) is to learn a disentangled representation of the variant causal factors that are related to environments and a policy π that depends on these factors. Our aim is to achieve OOD generalization for a policy π.

3.1 LEARNING REPRESENTATIONS FOR CAUSAL FACTORS

To learn a disentangled representation, we adopt a weakly-supervised generative model, which uses $q_\phi(z|s)$ to approximate the actual posterior distribution $p(z|s)$ by the variational inference (Higgins et al., 2017). By imposing shared constraints on partial approximation posteriors of the paired states, the encoded vector contains all the information about factors while guaranteeing they are disentangled. Specifically, we can achieve the above goal by optimizing the reconstruction loss

$$\begin{aligned}\max_{\phi,\psi} \mathbb{E}_{(s_x,s_y)}(\mathbb{E}_{\tilde{q}_\phi(z|s_x)} \log(p_\psi(s_x|z)) + \mathbb{E}_{\tilde{q}_\phi(z|s_y)} \log(p_\psi(s_y|z)) \\ +\beta D_{KL}(\tilde{q}_\phi(z|s_x)||p(z)) + \beta D_{KL}(\tilde{q}_\phi(z|s_y)||p(z))),\end{aligned} \tag{3}$$

where $\beta \geq 1$ and $\tilde{q}_\phi$ denotes that a subset of dimensions in q_ϕ is subject to the shared constraint, i.e., $q_\phi(z_i|s_x)$ and $q_\phi(z_i|s_y)$ are replaced by their average distribution (Locatello et al., 2020), where the dimension i is imposed the shared constraint. However, directly disentangling from the paired states is unreasonable because causal factors may satisfy both non-shared and shared constraints depending on the environmental setting. For example, in a grasping task, the robot may encounter some environments with different objects. If paired states belong to the same environment, the mass factor should align with the shared constraints; otherwise, it should align with the non-shared constraints.

Thus, in order to ensure that the causal factors align with the correct constraints, we need to satisfy the following conditions in two stages: (1) z should be invariant in the single environment, and (2) z should be varied across the different environments.

To fulfill condition (1), we propose a structure for the variational family that allows us to tractably perform approximate inference on the weakly-supervised generative model (see the pre-training stage in Figure 2(b)). In a single environment, the alignment constraints imposed by the paired states imply the true posterior: if $z_i \in \tilde{z}$, $p(z_i|s_x) \neq p(z_i|s_y)$; otherwise $p(z_i|s_x) = p(z_i|s_y)$. Inspired by Locatello et al. (2020), we enforce these constraints on the approximate posterior $q_\phi(z|s)$ of our learned model. Concretely, the constraints can be imposed by replacing each shared coordinate with the average (avg) of the two posteriors

$$\begin{aligned}\tilde{q}_\phi(z_i|s_x) &= q_\phi(z_i|s_x) && \textbf{if } i \leq k, \\ \tilde{q}_\phi(z_i|s_x) &= avg(q_\phi(z_i|s_x), q_\phi(z|s_y)) && \textbf{else}.\end{aligned} \tag{4}$$

$\tilde{q}_\phi(z|s_y)$ can be computed in an analogous manner. By implementing a D_{KL}-term to impose the hard constraint (4), $q_\phi(z|s_x)$ and $q_\phi(z|s_y)$ can jointly encode only one value per shared coordinate.

Out of all possible representations that are invariant in the individual environment, we want to search for one that also varies with the different environments, fulfilling condition (2). To ensure that the representations are environment-specific, we utilize the outcome of $\tilde{q}_\phi(z|s)$ to discover the subset of causal factors $\hat{z}$. Specially, we fine-tune the variational model learned from the previous stage to disentangle the latent features $\hat{z}$ via a new constraint over states collected from multiple environments (see the fine-tuning stage in Figure 2(b)). For the causal factors $[\hat{z}, \overline{z}]$, the alignment constraints imposed by the paired states imply another true posterior: if $z_i \in \hat{z}$, $p(z_i|s_y) \neq p(z_i|s_w)$; otherwise $p(z_i|s_y) = p(z_i|s_w)$. Similarly to Equation 4, the approximate posteriors are subject to the following constraints

$$\begin{aligned}\tilde{q}_\phi(z_i|s_y) &= q_\phi(z_i|s_y) && \textbf{if } i \leq k+m, \\ \tilde{q}_\phi(z_i|s_y) &= avg((q_\phi(z_i|s_y), q_\phi(z|s_w)) && \textbf{else}.\end{aligned} \tag{5}$$

$\tilde{q}_\phi(z|s_w)$ can be computed in an analogous manner. With the above two-stage constraints, we can separate $\hat{z}$ from other factors, where $\hat{z}$ is mapped into the dimensions range $[k+1:k+m]$ of the encoded vector. In particular, $\tilde{q}_\phi$ is only for the representation learning, and q_ϕ is then used for the policy's inputs.

3.2 TRAINING AGENTS FOR OOD GENERALIZATION

On the basis of the representation, we shall learn a generalizable policy π (parameterized by θ_π) in the multiple training environments, such that it maximizes the cumulative external rewards. To begin, we condition π on the representation $q_\phi(z|s)$ and adopt SAC as a basic optimization algorithm (Haarnoja et al., 2018). In each iteration, SAC performs the soft policy evaluation step and the

improvement step. The policy evaluation step updates a parametric Q-function $Q(s_t, a_t)$ by minimizing the soft Bellman residual. Then, the policy improvement step minimizes the KL divergence between the policy and a Boltzmann distribution induced by the Q-function

$$\mathcal{L}_\pi(\theta_\pi) = \mathbb{E}_{s_t \sim D}[D_{KL}(\pi(\cdot|q_\phi(z|s_t); \theta_\pi)||\mathcal{Q}(s_t, \cdot))], \tag{6}$$

where $\mathcal{Q}(s_t, \cdot)$ is positively related to $exp\{\frac{1}{\alpha}Q(s_t, \cdot)\}$. However, the disentangled representation learning tends to constrain all encoded factors to remain independent, which may slightly alter their semantic implication (Zhang et al., 2021c; Träuble et al., 2022). As a result, the input containing only disentangled factors may hurt generalization (see Figure 7). Instead, we learn a policy that establishes the dependence of actions on the partial representations $\mathcal{F}(z; z \sim q_\phi(z|s))$ by an additional network. $\mathcal{F}$ is a selector that captures the variant causal factors $\hat{z}$ by selecting features from k to $k + m$ dimensions, namely, $\mathcal{F}(z; z \sim q_\phi(z|s)) = z_{k+1:k+m}$. For brevity, we denote $\mathcal{F}(z; z \sim q_\phi(z|s))$ with $\mathcal{F}(q_\phi(z|s))$.

As shown in Figure 2(c), we add two separate networks (θ_1 and θ_2) to receive the state and the representations of variant causal factors separately in the policy model $\pi(\cdot|f_{\theta_1}(\mathcal{F}(q_\phi(z|s))), f_{\theta_2}(s); \theta_\pi, \theta_1, \theta_2)$. Then, in the policy improvement step, we optimize the following object

$$\mathcal{L}_\pi(\theta_\pi, \theta_1, \theta_2) = \mathbb{E}_{s_t \sim D} D_{KL}(\pi(\cdot|f_{\theta_1}(\mathcal{F}(q_\phi(z|s))), f_{\theta_2}(s); \theta_\pi, \theta_1, \theta_2)||\mathcal{Q}(s_t, \cdot)), \tag{7}$$

where f_{θ_1} is used to establish the dependence of actions on causal factors and f_{θ_2} is used to receive the information of the original state. In particular, the representation ϕ is frozen at this training stage.

Since the combination of causal inference and machine learning has been proven to be one of the effective ways to achieve OOD generalization (Cui & Athey, 2022), we further want to theoretically analyze whether the established dependence can recover the actual causal relationship. According to the definition of causality (Peters et al., 2017), learning the causal relationship between causal factors and actions needs to satisfy the condition: causal factors are independent of other factors that lead to actions. Inspired by Mahajan et al. (2021), we claim the following (proof in Appendix B).

Proposition 1 *Under the Causal MDP setup as above, if policy π converges to optimal π^*, where policy π^* can maximize the expected cumulative reward for all training environments E, then the GCRL for learning representations yields a generalizable agent such that learnt representation $\mathcal{F}(q_\phi(z|s))$ is independent of other factors in the controllable state s^c given the causal factors s^u. Specifically, the entropy $H(s^c|s^u) = H(s^c|\mathcal{F}(q_\phi(z|s)), s^u)$.*

The above proposition indicates that the representation of causal factors is independent of other factors in the states sampled by the optimal policy π^*. Thus, while the optimal policy can be found in training environments, the established dependence can recover the causal relationship between causal factors and actions. Meanwhile, the causal relationship ensures that the policy can achieve OOD generalization.

Further details and the full algorithm for optimizing GCRL can be found in Appendix A.

4 Experiments

The experimental studies aim at examining if the GCRL achieves the OOD generalization, and whether the causal relationships between causal factors and actions are recovered.

4.1 Experimental Setup

To verify the performance of the OOD generalization, we carry out benchmark tasks on Causal World (Ahmed et al., 2020) and Mujoco Control Suite (Tassa et al., 2018). In particular, additional experiments on OOD generalization, visualization, ablation, and parameter sensitivity can be found in Appendix C. Further details about the GCRL setting are provided in Appendix D.

Causal World The environments in the Causal World consist of a 3-fingered robot with 3 joints on each finger. Each environment is constrained to consist of a single object with which the agent interacts. For each object, the causal factors are mass (m), size (s), and height (h), which can be manipulated in the experiments.

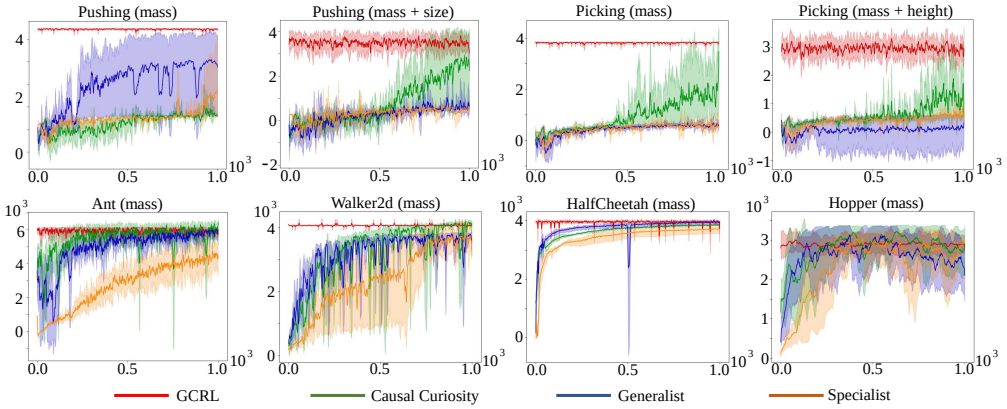


Figure 3: Comparison of OOD generalization in testing environments. The x-axis represents the episodes of extra training in the testing environments. The y-axis represents the average reward at each evaluation. GCRL is only learned in the training environments and deployed directly in the testing environments, which is the strict OOD generalization setting. The reward curve shows that GCRL generalizes better to the new values of causal factors compared with other baselines.

Mujoco Control Suite Mujoco includes a set of baseline environments for RL where the agent is required to simulate animal or human behavior. We select four verification environments (i.e., Ant, HalfCheetah, Hopper, and Walker), where their causal factors are the mass of the agent.

Baselines We compare the recent method Causal Curiosity (CC), which discovers binary representations of causal factors in different environments (Sontakke et al., 2021). Based on (Perez et al., 2020), we implement another two baselines, i.e., the Generalist and the Specialist, to verify the generalization ability of the GCRL. The Specialist does not access the training environments but is directly learned in the testing environments. Besides, the Specialist also serves as a benchmark for the complexity of the tasks. In contrast, the Generalist has access to the training environments but without access to the representations of causal factors. Finally, we add a pre-trained VAE baseline by following the setting of (Träuble et al., 2022). All comparison algorithms are implemented based on the SAC model to ensure fairness (Haarnoja et al., 2018).

4.2 Verification of OOD Generalization

To verify the OOD generalization performance of GCRL in unseen environments, we select multiple scenarios in the Causal World and Mujoco, where the settings of each scenario are consistent with (Sontakke et al., 2021). For Causal World, we select Pushing and Picking, and set different causal factors (mass, size, height) to create training environments and testing environments. Specifically, we set the mass from 0.01 to 0.05, the size from 2.0 to 2.5, and the height from 0.1 to 0.15 in the training environments. In testing environments, we increase the difficulty of the tasks by modifying the mass to 0.1, size to 1.5, and height to 0.2. For Mujoco, we set the mass from 0.5x to 1.0x the default in training environments, and 1.5x the default in testing environments. In particular, our GCRL only learns in the training environments, but other methods additionally access extra 1000 episodes in the testing environment.

Figure 3 shows the OOD generalization performance comparison of the algorithms in the testing environment, where the x-axis represents the extra data used by baselines. In general, we find that GCRL generalizes well to all unseen environments and outperforms other baselines without extra training. Furthermore, GCRL outperforms other algorithms that learn extra in testing environments from the Causal World. In particular, we noticed that two scenarios (Pushing and Picking) with augmented causal factors are challenging, which makes other baselines unable to achieve optimal through extra data. In Ant and Walker, the CC slightly outperforms us after extra training in the testing environments, but the ability of zero-shot generalization is significantly behind the GCRL. Finally, we summarize the rewards obtained by the OOD generalization policies and OOD adaptation policies in Table 1, where the rewards in Mujoco are multiplied by a coefficient of 10^{-3}.

Table 1: Performance comparison of OOD generalization policies and OOD adaptation policies in unseen environments. Values are episodic rewards averaged over 100 episodes using a held-out set of environment seeds. GCRL, CC, and Generalist are only learned in the training environments and deployed directly in the testing environments. CC°, Generalist°, and Specialist are baselines for extra training in the testing environments.

Environments	OOD generalization			OOD adaptation		
	GCRL	CC	Generalist	CC°	Generalist°	Specialist
Pushing-m	**4.44** ± **0.04**	0.02 ± 0.24	0.10 ± 0.53	0.62 ± 0.10	2.94 ± 1.79	1.62 ± 1.59
Pushing+m+s	**3.49** ± **0.42**	0.17 ± 0.30	0.67 ± 0.57	2.66 ± 1.55	0.63 ± 0.37	0.49 ± 0.15
Picking+m	**3.82** ± **0.04**	0.24 ± 0.18	0.06 ± 0.39	1.97 ± 1.17	0.60 ± 0.15	0.58 ± 0.22
Picking+m+h	**2.94** ± **0.40**	0.22 ± 0.19	0.13 ± 0.45	1.27 ± 0.96	0.16 ± 0.59	0.64 ± 0.22
Ant+m	5.93 ± 0.23	3.36 ± 1.50	2.48 ± 1.69	**6.03** ± **0.46**	5.91 ± 0.16	4.48 ± 0.78
Walker2d+m	4.07 ± 0.02	0.75 ± 0.04	1.03 ± 0.50	**4.11** ± **0.12**	3.75 ± 0.12	3.56 ± 0.56
Hopper+m	**2.89** ± **0.26**	1.85 ± 0.79	1.23 ± 0.72	2.73 ± 0.59	2.33 ± 0.64	2.87 ± 0.12
HalfCheetah+m	**3.95** ± **0.06**	2.71 ± 0.24	2.75 ± 0.27	3.86 ± 0.02	3.94 ± 0.07	3.72 ± 0.12

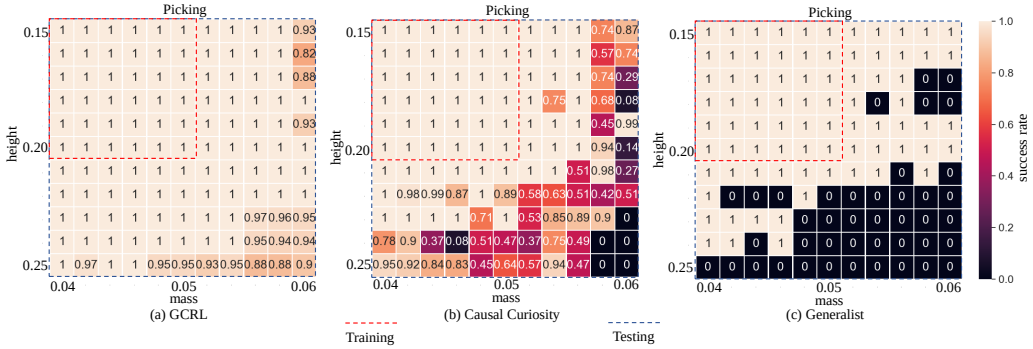
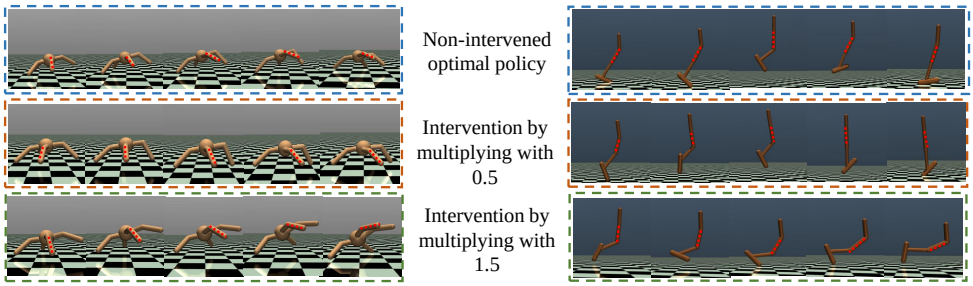


Figure 4: Comparison of OOD generalization in over 100 environments. The heatmap shows the success rate of GCRL, CC, and Generalist in each environment, where training environments are selected from the upper left quarter. GCRL can basically generalize to all difficult environments, while CC and Generalist can only generalize to some of the more similar environments.

Figure 4 shows the comparison of OOD generalization performance under more than one hundred groups of different causal factors in Picking, where the difficulty of the environment is proportional to the mass and height. Compared with CC and Generalist, GCRL achieves better generalization in all environments. This is because: 1) GCRL learns a continuous disentangled representation of causal factors from RL states, which can efficiently discover the important changes in environments; 2) GCRL recovers the causal relationship between actions and causal factors, ensuring that policies can directly predict the actions of unseen environments. The policies based on binary representation (CC) cannot generalize to factors with different values, resulting in poor performance in difficult environments. A broader range of the testing value can be seen in Appendix C.1.

4.3 Probing the Interpretability of GCRL

To better understand how the OOD generalization is achieved in GCRL, we intervene in disentangled representations and visualize the behavioral trajectories of policies in Figure 5. Based on the optimal policy, we record one optimal trajectory and two intervened trajectories separately in the Mujoco environments, where the intervened causal factor is the mass. For the representations of each causal factor, we introduce two interventions, i.e., grow and shrink, achieved by multiplying with a coefficient. This experiment allows us to observe the causal relationship between causal factors and actions. In Figure 5(a), we find that the intervention causes the ants to either fail to lift their legs or lift their legs excessively, compared with normal walking. This phenomenon intuitively illustrates GCRL's finding that mass affects leg strength, which is consistent with human understanding of their causal relationship. A similar phenomenon can be seen in Figure 5(b) that the hopper controls the



(a) Ant (b) Hopper

Figure 5: Visualization of behavioral trajectories under the interventions. The first trace shows the normal behaviors of the optimal policy without interventions. The last two trajectories show differences in behaviors after intervention on the representations (mass). It is worth noting that in this experiment, the coefficients act directly on the latent variables.

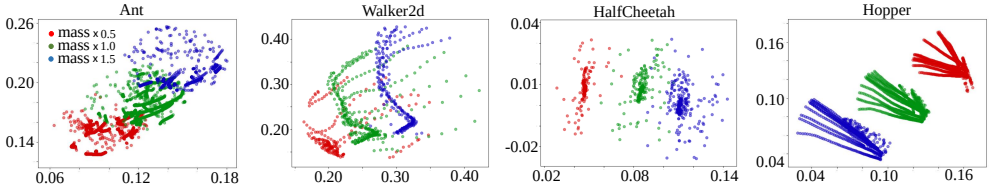


Figure 6: Visualization of the disentangled representations. The x-axis and y-axis correspond to the two dimensions of the encoded factor, respectively. The distributions of learned representations shift similarly with the setting of the masses, indicating that the information about variant causal factors is encoded into the specified dimensions.

jumps' force, resulting in two unbalanced trajectories due to the interventions. As a result, we find that GCRL is able to recover the causal relationship between causal factors and actions based on the learned representations, thereby generalizing to unseen environments.

4.4 Probing the Disentangled Representations of Causal Factors

To verify whether the learned representation satisfies the conditions presented in Section 3.1, we collect 30 trajectories generated in environments with different settings (mass) and visualize the low-dimensional embedding of the representations for the variant causal factors in the trajectories. For each task, we set the mass factor to be multiplied by three different coefficients (0.5, 1.0, 1.5) and assign a two-dimensional encoded vector to infer the approximate posterior. As shown in Figure 6, the shift of the distribution is approximated when the coefficients go from 0.5 to 1.0 and from 1.0 to 1.5. Furthermore, the distribution of the representation shifts similarly with the setting of the causal factor, indicating that it encodes the information of mass. These results further substantiate our proposal of the two-stage constraint in the GCRL framework.

4.5 Ablation Study

To verify the contribution of the disentanglement and each part of the representations to the generalization, we add an additional baseline pre-trained VAE (Träuble et al., 2022) and ablations that use each part of the representation as inputs to the policy. As shown in Figure 7(a), only the variant causal factor significantly improves the generalization, indicating that in the Causal MDP setting (Sontakke et al., 2021), factors that alter the dynamics of the environment need to be focused on. We find that the performance is slightly reduced when all factors are used as input to the policy. This is because some of these factors may be correlated with each other in the environment (Träuble et al., 2022). However, the disentangled representation learning tends to constrain the encoded factors to be as independent as possible, which may change their semantic implication (Träuble et al., 2022; Zhang et al., 2021c). In Figure 7(b), GCRL outperforms the pre-trained VAE, indicating that the weakly supervised disentanglement helps the policy model discover the critical changes in environmental dynamics and achieves better generalization.

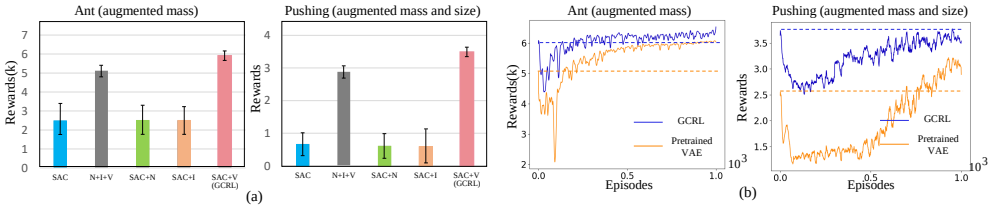


Figure 7: Ablation study on representations. (a) Ablations that use each part of the representation (N: non-causal factor, I: invariant causal factor, and V: variant causal factor), and all of them, as inputs to the policy. (b) Compared with the pre-trained VAE. The solid lines represent the fine-tuning curves; the dash lines represent the generalization rewards in testing environments.

5 Related Work

Causality in Reinforcement Learning Several ideas from causality have been applied to some aspects of reinforcement learning, such as sample efficiency (Seitzer et al., 2021; Wang et al., 2020), generalization (Lee et al., 2021; Yao et al., 2018), and theoretical analysis (Lee & Bareinboim., 2020; Zhang & Bareinboim., 2019). Sonar et al. (2021) utilize the Invariant Risk Minimization (Arjovsky et al., 2019; Ahuja et al., 2020; Tobin et al., 2017) to learn an optimal RL policy that generalizes across domains. To avoid spurious correlations and learn the actual causal relationship, Zhang et al. (2020); Bica et al. (2021) propose the idea of conditioning the policy on causal parents that are shared across environments. The above methods have been well verified in scenarios with different task-irrelevant features, such as different backgrounds. In contrast, we try to explore another scenario with different task-relevant features, such as manipulating a robot to move boxes of different sizes and qualities. Perhaps a similar setting to GCRL is one in (Sontakke et al., 2021), which proposes to discover the causal factors in multiple environments and then equip the policy to improve generalization. However, their models learn the OOD adaptation policy, leading to a limited generalization that requires extra training data from the testing environments.

Representation in Reinforcement Learning Various existing deep RL methods have been proposed to learn representations implicitly by optimizing some RL objectives (Zhang et al., 2021a; Sodhani et al., 2021; Yarats et al., 2021; Träuble et al., 2022; Moskovitz et al., 2022; Chen & Pan., 2022; Touati & Ollivier., 2021; Yamada et al., 2022; Träuble et al., 2021; Wahlstrom et al., 2015). Unlike these common methods, disentangled representations are more interpretable (Hosoya, 2019) and are also used in RL algorithms (Liu et al., 2021). However, Locatello et al. (2019) theoretically prove that unsupervised disentanglement without inductive biases is impossible and highly unstable, susceptible to random seed values. To address this issue, they also theoretically prove that pairwise inputs provide sufficient inductive bias to disentangle causal factors of variation (Locatello et al., 2020). Unfortunately, the above method cannot be directly applied in our setting because causal factors may satisfy both non-shared and shared constraints. Instead, we split the process of disentangling into two stages: the pre-training stage in a single environment and the fine-tuning stage in multiple environments.

6 Conclusion

In this paper, we propose a novel framework GCRL for learning RL agents that achieve the OOD generalization. In GCRL, we propose an encoder that learns a disentangled representation from paired states with a two-stage constraint, which can capture information about causal factors. Based on the disentangled representations, we learn a policy that establishes a dependence of action on causal factors by setting an independent network. Moreover, we theoretically prove that the established dependence can recover the causal relationship between causal factors and actions when the policy approaches optimality. Experimental results verify that agents learned by our framework can generalize to unseen environments from Causal World and Mujoco. The visualized results show that after intervening with representations, our model is able to generate semantic behaviors based on the recovered causal relationship.

REFERENCES

O Ahmed, F Träuble, A Goyal, A Neitz, M Wüthrich, Y Bengio, B Schölkopf, and S. Bauer. Causalworld: A robotic manipulation benchmark for causal structure and transfer learning. *arXiv peprint arXiv:2010.04296*, 2020.

Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. In *International Conference on Machine Learning (PMLR)*, pp. 145–155, 2020.

K Akuzawa, Y Iwasawa, and Y. Matsuo. Adversarial invariant feature learning with accuracy constraint for domain generalization. In *In Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 315–331, 2019.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv peprint arXiv:1907.02893*, 2019.

Ioana Bica, Daniel Jarrett, and Mihaela van der Schaar. Invariant causal imitation learning for generalizable policies. In *Advances in Neural Information Processing Systems (NIPS)*, 2021.

Jianda Chen and Sinno Jialin Pan. Learning generalizable representations for reinforcement learning via adaptive meta-learner of behavioral similarities. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment inference for invariant learning. In *International Conference on Machine Learning (PMLR)*, pp. 2189–2200, 2021.

Peng Cui and Susan Athey. Stable learning establishes some common ground between causal inference and machine learning. *Nature Machine Intelligence*, 4(2):110–115, 2022.

Finale Doshi-Velez and George Konidaris. Hidden parameter markov decision processes: A semiparametric regression approach for discovering latent task parametrizations. In *IJCAI: proceedings of the conference*, volume 2016, pp. 1432. NIH Public Access, 2016.

T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning (PMLR)*, pp. 1861–1870, 2018.

I Higgins, L Matthey, A Pal, C Burgess, X Glorot, M Botvinick, S Mohamed, and A. Lerchner. betavae: Learning basic visual concepts with a constrained variational framework. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

Haruo Hosoya. Group-based learning of disentangled representations with generalizability for novel contents. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2506–2513, 2019.

Biwei Huang, Fan Feng, Chaochao Lu, Sara Magliacane, and Kun Zhang. Adarl, what, where, and how to adapt in transfer reinforcement learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

Taylor W Killian, Samuel Daulton, George Konidaris, and Finale Doshi-Velez. Robust and efficient transfer learning with hidden parameter markov decision processes. *Advances in neural information processing systems*, 30, 2017.

Robert Kirk, Amy Zhang, Edward Grefenstette, and Tim Rocktäschel. A survey of generalisation in deep reinforcement learning. *arXiv peprint arXiv:2111.09794*, 2021.

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pp. 5815–5826. PMLR, 2021.

Sanghack Lee and Elias Bareinboim. Characterizing optimal mixed policies: Where to intervene and what to observe. In *Advances in Neural Information Processing Systems (NIPS)*, 2020.

Timothy E Lee, Jialiang Zhao, Amrita S Sawhney, Siddharth Girdhar, and Oliver Kroemer. Causal reasoning in simulation for structure and transfer learning of robot manipulation policies. *arXiv peprint arXiv:2103.16772*, 2021.

Guiliang Liu, Xiangyu Sun , Oliver Schulte, and Pascal Poupart. Learning tree interpretation from object representation for deep reinforcement learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2021.

F Locatello, S Bauer, M Lucic, S Gelly, B Schölkopf, and O. Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In *International Conference on Machine Learning (PMLR)*, 2020.

Divyat Mahajan, Shruti Tople, and Amit Sharma. Domain generalization using causal matching. In *International Conference on Machine Learning (PMLR)*, 2021.

Russell Mendonca, Xinyang Geng, Chelsea Finn, and Sergey Levine. Meta-reinforcement learning robust to distributional shift via model identification and experience relabeling. *arXiv preprint arXiv:2006.07178*, 2020.

Ted Moskovitz, Spencer R. Wilson, and Maneesh Sahani. A first-occupancy representation for reinforcement learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

J. Pearl. *Causality.* Cambridge university press, 2009.

C. F Perez, F. P Such, and T. Karaletsos. Generalized hidden parameter mdps: Transferable model-based rl in a handful of trials. In *Thirty-fourth AAAI conference on artificial intelligence*, 2020.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms.* MIT Press, 2017.

Maximilian Seitzer, Bernhard Schölkopf, and Georg Martius. Causal influence detection for improving efficiency in reinforcement learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2021.

Zheyan Shen, Jiashuo Liu, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*, 2021.

Rui Shu, Yining Chen, Abhishek Kumar, Stefano Ermon, and Ben Poole. Weakly-supervised disentanglement with guarantees. In *International Conference on Robot Learning (ICRL)*, pp. 1094–1100, 2020.

D Silver, J Schrittwieser, K Simonyan, I Antonoglou, A Huang, A Guez, T Hubert, L. R Baker, M Lai, A Bolton, Y Chen, T. P Lillicrap, F Hui, L Sifre, G van den Driessche, T Graepel, , and D. Hassabis. Mastering the game of go without human knowledge. *Nature*, 550:354–359, 2017.

Shagun Sodhani, Amy Zhang, and Joelle Pineau. Multi-task reinforcement learning with context-based representations. In *International Conference on Machine Learning (PMLR)*, 2021.

Anoopkumar Sonar, Vincent Pacelli, and Anirudha Majumdar. Invariant policy optimization: Towards stronger generalization in reinforcement learning. In *Learning for Dynamics and Control*, pp. 21–33. PMLR, 2021.

Sumedh A Sontakke, Arash Mehrjou, Laurent Itti, and Bernhard Schölkopf. Causal curiosity: Rl agents discovering self-supervised experiments for causal representation learning. In *International Conference on Machine Learning (PMLR)*, 2021.

R. S. Sutton and A. G. Barto. Reinforcement learning: An introduction. *MIT press*, 1998.

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

J Tobin, R Fong, A Ray, J Schneider, W Zaremba, and P. Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *International Conference on Intelligent Robots and Systems (IROS)*, pp. 23–30, 2017.

Ahmed Touati and Yann Ollivier. Learning one representation to optimize all rewards. In *Advances in Neural Information Processing Systems (NIPS)*, 2021.

Frederik Träuble, Andrea Dittadi, Manuel Wüthrich, Felix Widmaier, Peter Gehler, Ole Winther, Francesco Locatello, Olivier Bachem, Bernhard Schölkopf, and Stefan Bauer. Representation learning for out-of-distribution generalization in reinforcement learning. In *International Conference on Machine Learning (PMLR)*, 2021.

Frederik Träuble, Andrea Dittadi, Manuel Wüthrich, Felix Widmaier, Peter Gehler, Ole Winther, Francesco Locatello, Olivier Bachem, Bernhard Schölkopf, and Stefan Bauer. The role of pretained representations for the ood generalization of rl agents. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

O Vinyals, I Babuschkin, W. M Czarnecki, M Mathieu, A Dudzik, J Chung, D. H Choi, R Powell, T Ewalds, and P. Georgiev. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 7782:350–354, 2019.

Niklas Wahlstrom, Thomas BSchon, and Marc Peter Deisenroth. From pixels to torques: Policy learning with deep dynamical models. *CoRR, abs/1502.02251*, 2015.

Lingxiao Wang, Zhuoran Yang, and Zhaoran Wang. Provably efficient causal reinforcement learning with confounded observational data. In *Advances in Neural Information Processing Systems (NIPS)*, 2020.

Jun Yamada, Karl Pertsch, Anisha Gunjal, and Joseph J. Lim. Task-induced representation learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

J Yao, T Killian, G Konidaris, and F. Doshi-Velez. Direct policy transfer via hidden parameter markov decision processes. *LLARLA Workshop, FAIM*, 2018, 2018.

Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Reinforcement learning with prototypical representations. In *International Conference on Machine Learning (PMLR)*, 2021.

A Zhang, R Mcallister, R Calandra, Y Gal, and S. Levine. Learning invariant representations for reinforcement learning without reconstruction. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021a.

A Zhang, S Sodhani, K Khetarpal, and J. Pineau. Learning robust state abstractions for hidden parameter block. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021b.

Amy Zhang, Clare Lyle, Shagun Sodhani, Angelos Filos, Marta Kwiatkowska, Joelle Pineau, Yarin Gal, and Doina Precup. Invariant causal prediction for block mdps. In *International Conference on Machine Learning (PMLR)*, 2020.

J. Zhang and E. Bareinboim. Near-optimal reinforcement learning in dynamic treatment regimes. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 13401–13411, 2019.

Xingxuan Zhang, Peng Cui, Renzhe Xu, Linjun Zhou, Yue He, and Zheyan Shen. Deep stable learning for out-of-distribution generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5372–5382, 2021c.

Yufan Zhou, Zhenyi Wang, Jiayi Xian, Changyou Chen, and Jinhui Xu. Meta-learning with neural tangent kernels. *arXiv preprint arXiv:2102.03909*, 2021.

L Zintgraf, K Shiarlis, M Igl, S Schulze, Y Gal, K Hofmann, and S. Whiteson. Varibad: A very good method for bayes-adaptive deep rl via meta-learning. *arXiv peprint arXiv:1910.08348*, 2019.

Appendix

A PSEUDOCODE FOR GCRL

The pseudocode for GCRL training is shown in Algorithm 1. The pre-training stage of representation learning for causal factors corresponds to lines $2-13$ (Algorithm 1). The fine-tuning stage of representation learning for causal factors corresponds to lines $15-22$ (Algorithm 1). The last stage of policy learning for OOD generalization corresponds to lines $24-32$ (Algorithm 1). Source code is available at here.

Algorithm 1: Generalizable Causal Reinforcement Learning (GCRL)

Input: CausalMDP $G =< S, A, P, R, \gamma >$; The policy (parametered by θ_π) with two independent networks θ_1, θ_2; The disentangled representation model q_ϕ, A set of training environments $E = \{e_i\}_{i=1}^n$

Output: The learnt policy $\theta_\pi, \theta_1, \theta_2$; The learnt disentangled representation model q_ϕ

1 Initialize $\theta_\pi, \theta_1, \theta_2, \phi$
2 Pre-training q_ϕ in single environment e_i
3 **for** *each iteration* **do**
4 Sampling from a single environment e_i
5 Update representation model q_ϕ with weakly-supervised method
6 Take a minibatch paired states from the replay buffer
7 Add the following constrains to q_ϕ
8 $\tilde{q}_\phi(z_i|s_x) = q_\phi(z_i|s_x)$ **if** $i \leq k$
9 $\tilde{q}_\phi(z_i|s_x) = avg(q_\phi(z_i|s_1x) + q_\phi(z|s_y))$ **else**
10 Update q_ϕ by maximizing the ELBO object
11 $\max_{\phi,\theta} \mathbf{E}_{(s_1,s_2)} \mathbf{E}_{\tilde{q}_\phi(z|s_1)} \log(p_\theta(s_1|z)) + \mathbf{E}_{\tilde{q}_\phi(z|s_2)} \log(p_\theta(s_2|z))$
12 $+\beta D_{KL}(\tilde{q}_\phi(z|s_1)||p(z)) + \beta D_{KL}(\tilde{q}_\phi(z|s_2)||p(z))$
13 **end**
14 Refresh the replay buffer
15 Fine-tuning q_ϕ in multiple environments E
16 **for** *each iteration* **do**
17 Sampling from multiple environments E
18 Add the following constrains to q_ϕ
19 $\tilde{q}_\phi(z_i|s_y) = q_\phi(z_i|s_y)$ **if** $i \leq k + m$
20 $\tilde{q}_\phi(z_i|s_y) = avg(q_\phi(z_i|s_y) + q_\phi(z|s_w))$ **else**
21 Fine-tuning q_ϕ by maximizing the ELBO object
22 **end**
23 Refresh the replay buffer
24 Training $\theta_\pi, \theta_1, \theta_2$ with q_ϕ
25 **for** *each iteration* **do**
26 Sampling from multiple environments E
27 **for** *each epoch* **do**
28 Take a minibatch data from the replay buffer
29 Update a parametric Q-function $Q(s_t, a_t)$ by minimizing the soft Bellman residual
30 Update agents by minimizing the following object
31 $\mathcal{L}_\pi(\theta_\pi, \theta_1, \theta_2) = \mathbf{E}_{s_t \sim D}[D_{KL}(\pi(\cdot|f_{\theta_1}(\mathcal{F}(q_\phi(z|s))), f_{\theta_2}(s); \theta_\pi)||\mathcal{Q}(s_t, \cdot))]$
32 **end**
33 **end**

B PROOF OF PROPOSITION 1

Proposition 1 relates to the Causal MDP setup, where the state contains causal and no-causal factors. This setting allows different environments have different causal factors s^u, leading to different $P(s)$ marginals. Similar to the Environment Invariance Contraint (Creager et al., 2021), we assume that

the controllable states s^c in different environments are the same for the task. In other words, there exists some s^c such that the $p(s^c)$ marginals should be the same (as well as $p(s^c|\pi^*)$). The proof uses the entropy formulation of distribution-matching methods, as done by (Akuzawa et al., 2019).

Proof. We can write the GCRL model as optimizing two objectives: maximize the expected rewards in the multiple training environments, and learn a disentangled representation $\mathcal{F}(\tilde{q}_\phi(z|s))$ that is specific to the environment e given the optimal policy π^*. For simplicity, we use $\mathcal{F}(s)$ refer to $\mathcal{F}(\tilde{q}_\phi(z|s))$ in the following.

Let us focus on the second objective, which can be interpreted as maximizing the entropy of other factors in the controllable state s^c given the optimal policy π^* and disentangled representation $H(s^c|\pi^*, \mathcal{F}(s))$. Let $\mathcal{F}^*$ be the optimal representation for the GCRL. Then, we can write

$$\mathcal{F}^* = \arg\max_{\mathcal{F}} H(s^c|\pi^*, \mathcal{F}(s)). \tag{8}$$

Based on the property of entropy, we have $H(s^c|\pi^*, \mathcal{F}(s)) \leq H(s^c|\pi^*)$. Thus, the optimal $\mathcal{F}^*$ satisfies

$$H(s^c|\pi^*, \mathcal{F}^*(s)) = H(s^c|\pi^*). \tag{9}$$

Now two cases arise: $s^u \perp\!\!\!\perp s^c|\pi^*$ or $s^u \not\perp\!\!\!\perp s^c|\pi^*$. Based on the Environment Invariance Contraint, we assume the former. If s^u is independent of other factors in the controllable state s^c conditioned on the optimal policy, then

$$H(s^c|\pi^*) = H(s^c|\pi^*, s^u). \tag{10}$$

Here environment s^c is independent of both s^u and $\mathcal{F}^*(s)$, conditional on π^*. Since the causal factors s^u cannot be caused by the representation $\mathcal{F}^*(s)$, it cannot be a collider in any graph connecting s^c, s^u and $\mathcal{F}^*(s)$ (Pearl, 2009). Thus, conditioning on s^u does not remove the independence between s^c and $\mathcal{F}^*(s)|\pi^*$ (conditioned on π^*). Hence, we condition on Eq. 9 with s^u and obtain

$$H(s^c|\pi^*, s^u) = H(s^c|\pi^*, s^u, \mathcal{F}^*(s)). \tag{11}$$

Plugging it into the above equations, we drive

$$H(s^c|\pi^*) = H(s^c|\pi^*, s^u) = H(s^c|\pi^*, s^u, \mathcal{F}^*(s)). \tag{12}$$

Since there is a unique optimal solution for the environment, s^u can completely predict the optimal policy π^*. Thus, we can remove π^* from the above equation

$$H(s^c|s^u) = H(s^c|\mathcal{F}^*(s), s^u). \tag{13}$$

The above equation implies that the learned representation $\mathcal{F}^*(s)$ is independent of other factors in the controllable state s^c given the causal factors s^u. Thus $\mathcal{F}^*(s)$ depends on s^u and not on any other factors.

C Additional Experiments

C.1 Additional Results of Generalization over Causal World

We additionally verify the OOD generalization performance of GCRL in Pushing and add the comparison results of the Generalist. In the training environments, we set the mass from 0.1 to 0.15 and the size from 2.0 to 2.25. In the test, we set different environments by modifying the mass from 0.05 to 0.2 and size from 1.75 to 2.25. Figure 8 shows the comparison of generalization performance under more than one hundred groups of different causal factors in Pushing and picking, where the difficulty of the environment is proportional to the quality and inversely proportional to the size. GCRL achieves better generalization than Causal Curiosity in all environments. This is because (1) GCRL can efficiently discover the important changes in environments by learning a continuous disentangled representation from RL states. (2) GCRL can predict new actions for unseen causal factors by establishing causality between actions and causal factors. Based on original states, the Generalist cannot adapt to factors with large differences, resulting in poor performance in unseen environments. In Figure 9, training environments are selected from the bottom right quarter. Due to the simpler dynamics of the environment in the test distribution, models are easier to achieve the zero-shot generalization.

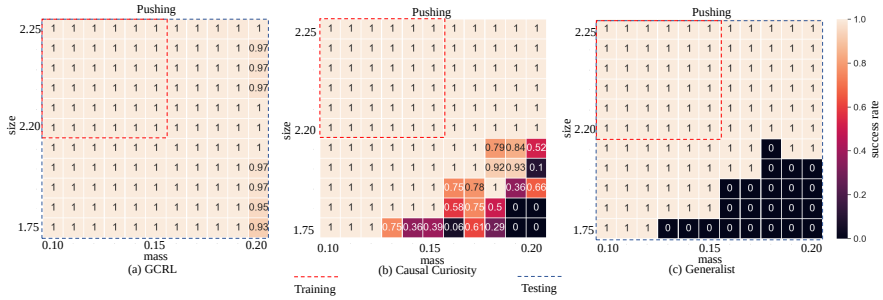


Figure 8: Comparison of OOD generalization in over one hundred environments. The heatmap shows the success rate of GCRL, Causal Curiosity, and Generalist in each environment, where training environments are selected from the upper left quarter. Generalist has the worst generalizability because it is the only method that does not consider causal factors. GCRL can basically generalize to all difficult environments, while Causal Curiosity can only generalize to some of the more similar environments.

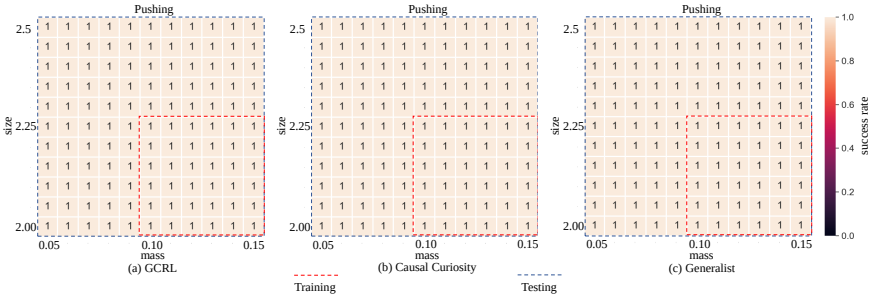


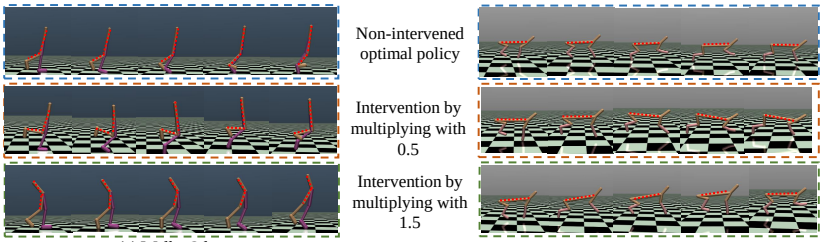
Figure 9: Comparison of OOD generalization in over one hundred environments. The heat map shows the success rate of GCRL, Causal Curiosity in each environment, where training environments are selected from the bottom right quarter. Benefiting from the simpler dynamics of the environment in the test distribution, the model is easier to achieve the zero-shot generalization.

C.2 Additional Visualization Results of Mujoco

We also record one non-intervention and two intervention trajectories separately in another two Mujoco environments, where the intervened factor is the mass. In Figure 10 (a), we found that both forward and back occurred after the intervention compared to normal walking. This phenomenon intuitively illustrates GCRL's finding that mass affects leg strength, which is consistent with human understanding of their causal relationship. A similar phenomenon can be seen in Figure 10 (b) that the cheetah controls the jumps' force according to the mass, resulting in two unbalanced trajectories due to the interventions. As a result, we find that GCRL is able to discover the actual causal relationship between causal factors and actions based on learned representations in training sets, thereby generalizing to unseen environments.

C.3 Parameter Sensitivity Analysis

We select the β, k, and m to evaluate whether our model is sensitive to specific essential hyperparameters. The results are shown in Figure 11. For β, we find that $\beta = 1$ is enough for GCRL to disentangle the causal factors, leading to excellent generalization. For k, we recommend choosing a relatively larger value ($> m$) to ensure complete encoding of controllable states, which is beneficial to disentangle causal factors of variation in the fine-tuning stage. For m, the results in Figure 11 (c) show that its value is inversely proportional to the generalization performance of the policy because the large m may lead to an incorrect encoding of non-causal factors. As a result, We conclude as follows: (1) The choice of β does not significantly affect the performance of the model. (2) A large k facilitates the separation of non-causal factors. (3) A small m is sufficient to encode causal factors, which is consistent with the conclusion in (Hosoya, 2019).



(a) Walker2d (b) HalfCheetah

Figure 10: Visualization of behavioral trajectories under the interventions. The first trace shows the normal behaviors of the optimal policy without interventions. The last two trajectories show differences in behaviors after intervention on the representations (mass).

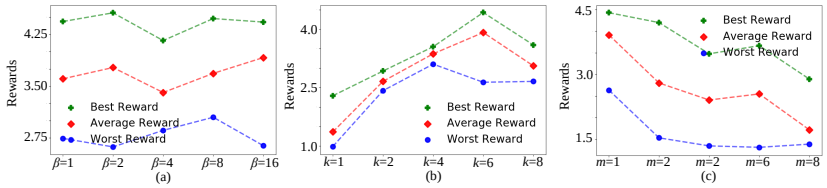


Figure 11: Parameter Sensitivity Analysis. (a) Performance comparisons based on constraints of different strengths β. (b) Performance comparisons of different encoding dimensions k for non-causal factors. (c) Performance comparisons of different encoding dimensions m for causal factors.

C.4 Additional Ablation Study

Our goal is to discover the dependencies of actions on causal factors. To facilitate the model to achieve this goal, we add additional parameters θ_1 to receive disentangled representations of the variant causal factors and θ_2 to receive the original state. To verify the advantage of using additional networks to learn the dependence of actions on causal factors proposed in Section 3.2, we compare with the original policy networks (θ_π) that receive both the representations and states. As shown in Figure 12, GCRL achieves better performance since the additional network is able to increase the influence of causal factors on the choice of actions. In the Hopper scenario, the original policy exhibits poor generalization and is significantly inferior to our policy. This is because the mass factor is not obvious and invariant in the same environment, leading to the importance of its features in generalization is not easily perceived by a single network.

C.5 The Reward Gap Between Training and Testing

Consistent with the settings in Figure 3, we report the training curves for GCRL and CC. As shown in Figure 13, both GCRL and CC have achieved good performance in the training environments. However, CC generalizes poorly in the testing environment, especially in the Causal World. This is because CC has not learned to adjust policies based on the changes in causal factors.

C.6 Environments with Different Task-irrelevant Factors

In real-world applications, task-irrelevant features such as background and noise may also change and induce spurious correlations. In this case, a common approach is to learn representations that are invariant across environments (Zhang et al., 2021a; 2020). Therefore, we need to separate this part of the features to ensure that only task-relevant features are used as the input of the policy. Fortunately, task-irrelevant features can also be disentangled by GCRL models using two-stage constraints because they usually depend on the setting of the environment, such as grasping an object with different colors (Träuble et al., 2022). In other words, task-irrelevant factors will be encoded into $\hat{z}$, while other parts are invariant across environments. We can define a selector $\widetilde{\mathcal{F}}$ to filter the task-relevant factors corresponding to $[\tilde{z}, \bar{z}]$ by selecting features from 1 to k and from $k+m$ to d dimensions, namely, $\widetilde{\mathcal{F}}(z; z \sim q_\phi(z|s)) = [z_{1:k}, z_{k+m:d}]$. Similar to Section 3.2, we define $\widetilde{\mathcal{F}}(q_\phi(z|s))$ to represent $\widetilde{\mathcal{F}}(z; z \sim q_\phi(z|s))$. Due to the presence of task-irrelevant factors in the

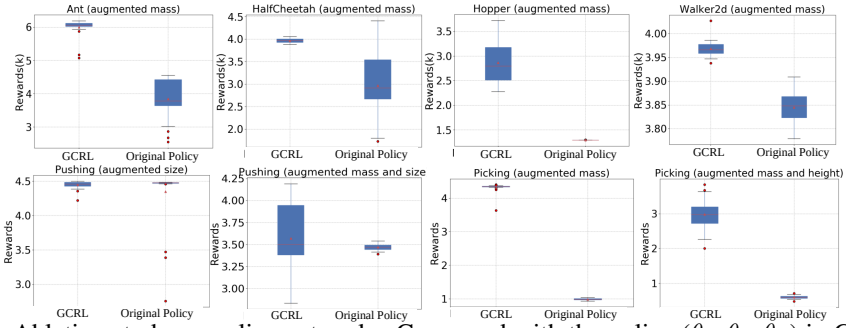


Figure 12: Ablation study on policy networks. Compared with the policy $(\theta_1, \theta_2, \theta_\pi)$ in GCRL, the original policy receive both representations of the variant causal factors and the original state by a single network (θ_π).

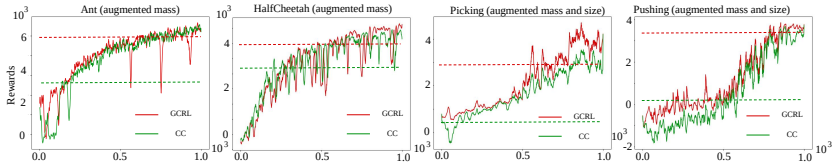


Figure 13: The reward gap between training and testing. The solid lines denote the reward cure in the training environments, and the dashed lines represent the generalization rewards in the test environments.

original state, we train the model to rely directly on the representation of task-relevant factors:

$$\mathcal{L}_\pi(\theta_\pi) = \mathbb{E}_{s_t \sim D}[D_{KL}(\pi(\cdot|\widetilde{\mathcal{F}}(q_\phi(z|s_t)); \theta_\pi)||\mathcal{Q}(s_t, \cdot))], \tag{14}$$

where $\mathcal{Q}(s_t, \cdot)$ is approximately computed by the a parametric Q-function in SAC, and the representations model ϕ is frozen at this training stage.

To verify the generalization of GCRL in the testing environment with different task-irrelevant factors, we set different colored objects in Causal World and different backgrounds in Mujoco. We test factors of different combination types as input to the policy and compare them with the original state. As shown in Figure 14, the combination of N and I achieve the best performance, indicating that the color information is encoded into the V part. The other two combinations ($N + V$ and $I + V$) suffer severe performance degradation due to the lack of task-relevant information.

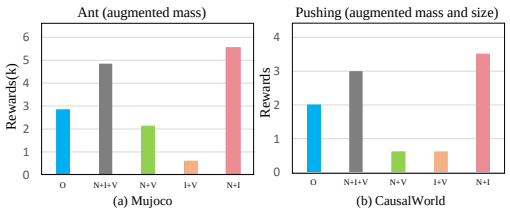


Figure 14: The OOD generalization testing in evironments with different task-irrelevant factors (O:original state, N: non-causal factor, I: invariant causal factor, and V: variant causal factor).

D Implement Details

We implement the disentangled representation model using VAE as the core component in combination with fully-connected layers (FC layer). Experiments are carried out on NVIDIA GeForce RTX 2070 SUPER GPUs and with fixed hyperparameter settings, as described in the following. For the encoder, we employ a nonlinear layer (256 units) based on the RELU activation function and an FC layer (16 units) to output the posterior distribution $q_\phi(z|s)$. we set the dimension z_dim of $q_\phi(z|s)$ is 16. For the decoder, we employ a nonlinear layer (16 units) based on the RELU activation function and an FC layer (the size is equal to states) to reconstruct the input from the encoder. For constraining, we refer to (Hosoya, 2019) to select $k = 6$ in the single environment, and m is equal to the number of factors that change in the multiple environments. k is relatively large to ensure that non-causal factors are adequately coded, while m is relatively small to avoid non-causal factors from

being coded in the fine-tuning stage. For training, we collect 1000 episodes in single environments and 100 episodes in multiple environments.

Table 2: Hyperparameters used in Causal World.

common hyperparameters	value
soft target tau	5e-3
gamma	0.95
learning rate in RL	1e-4
learning rate in representation	1e-4
batch size	256
optimizer	Adam
reward scale	1
max path length	1000
num trains per train loop	1000
use automatic entropy tuning	Ture

Table 3: Hyperparameters used in Mujoco.

common hyperparameters	value
soft target tau	5e-3
gamma	0.99
learning rate in RL	1e-4
learning rate in representation	3e-4
batch size	256
optimizer	Adam
reward scale	1
max path length	1000
num trains per train loop	1000
use automatic entropy tuning	Ture

We implement the policy using SAC as the core component in combination with two independent networks. For f_{θ_1}, we employ a nonlinear layer (32 units) based on the RELU activation function and an FC layer (16 units) to output the causal effect on actions. For f_{θ_2}, we employ a nonlinear layer (256 units) based on the RELU activation function and an FC layer (16 units) to predict the causal effect from controllable state. For the output of actions, we employ an FC layer (the number of units is equal to the action space in environments) to predict the final actions. For training, we collect 1000 episodes in training environments.

More hyperparameters for GCRL are shown in Table 2 and Table 3.