Causal Influence Detection for Improving Efficiency in Reinforcement Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Many reinforcement learning (RL) environments consist of independent entities that interact sparsely. In such environments, RL agents have only limited influence over other entities in any particular situation. Our idea in this work is that learning can be efficiently guided by knowing when and what the agent can influence with its actions. To achieve this, we introduce a measure of *situation-dependent causal influence* based on conditional mutual information and show that it can reliably detect states of influence. We then propose several ways to integrate this measure into RL algorithms to improve exploration and off-policy learning. All modified algorithms show strong increases in data efficiency on robotic manipulation tasks.

1 Introduction

Reinforcement learning (RL) is a promising route towards versatile and dexterous artificial agents. Learning from interactions can lead to robust control strategies that can cope with all the intricacies of the real world that are hard to engineer correctly. Still, many relevant tasks such as object manipulation pose significant challenges for RL. Although impressive results have been achieved using simulation-to-real transfer [1] or heavy physical parallelization [2], training requires countless hours of interaction. Improving sample efficiency is thus a key concern in RL. In this paper, we approach this issue from a causal inference perspective.

When is an agent in control of its environment? An agent can only influence the environment by its actions. This seemingly trivial observation has the underappreciated aspect that the causal influence of actions is *situation dependent*. Consider the simple scenario of a robotic arm in front of an object on a table. Clearly, the object can only be moved when contact between the robot and object is made. Generally, there are situations where immediate causal influence is possible, while in others, none is. In this work, we formalize this situation-dependent nature of control and show how it can be exploited to improve the sample efficiency of RL agents. To this end, we derive a measure that captures the causal influence of actions on the environment and devise a practical method to compute it.

Knowing when the agent has control over an object of interest is important both from a learning and an exploration perspective. The learning algorithm should pay particular attention to these situations because (i) the robot is initially rarely in control of the object of interest, making training inefficient, (ii) physical contacts are hard to model, thus require more effort to learn and (iii) these states are enabling manipulation towards further goals. But for learning to take place, the algorithm first needs data that contains these relevant states. Thus, the agent has to take its causal influence into account already during exploration.

We propose several ways in which our measure of causal influence can be integrated into RL algorithms to address both the exploration, and the learning side. For exploration, agents can be rewarded with a bonus for visiting states of causal influence. We show that such a bonus leads the agent to quickly discover useful behavior even in the absence of task-specific rewards. Moreover,

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

our approach allows to explicitly guide the exploration to favor actions with higher predicted causal impact. This works well as an alternative to ϵ-greedy exploration, as we demonstrate. Finally, for learning, we propose an off-policy prioritization scheme and show that it reliably improves the speed of learning. Each of our investigations is backed by empirical evaluations in robotic manipulation environments and demonstrates a clear improvement of the state-of-the-art with the same generic influence measure.

2 Related Work

The idea underlying our work is that an agent can only sometimes influence its surroundings. This rests on two basic assumptions about the causal structure of the world. The first is that the world consists of independent entities, in accordance with the principle of independent causal mechanisms (ICM) [3], stating that the world's generative process consists of autonomous modules. The second assumption is that the potential influence that entities have over other entities is localized spatially and occurs sparsely in time. We can see this as explaining the sparse mechanism shift hypothesis, which states that naturally occurring distribution shifts will be due to local mechanism changes [4]. This is usually traced back to the ICM principle, i.e. that interventions on one mechanism will not affect other mechanisms [5]. But we argue that it is also due to the *limited interventional range* of agents (or, more generally, physical processes), which restricts the breadth and frequency of mechanism-changes in the real world. Previous work has used sparseness to learn disentangled representations [6, 7], causal models [8], or modular architectures [9]. In the present work, we show that taking the localized and sparse nature of influence into account can also strongly improve RL algorithms.

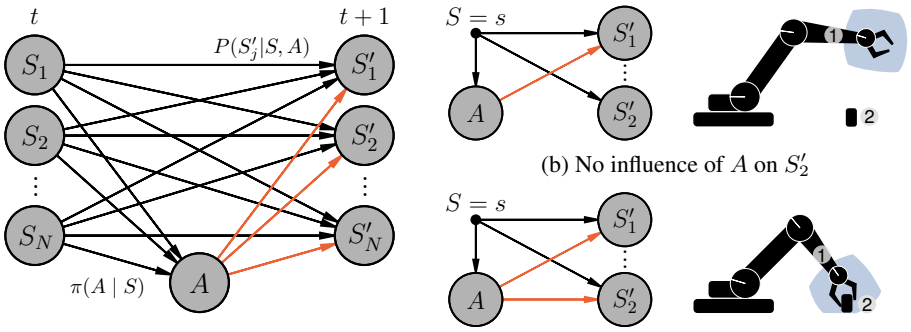
Detecting causal influence, informally, means deciding whether changing a causal variable would have an impact on another variable. This involves causal discovery, that is, finding the existence of arrows in the causal graph [10]. While the task of causal discovery is unidentifiable in general [11], there are assumptions which permit discovery [12], in particular in the time series setting we are concerned with [13]. Even if the existence of an arrow is established, the problem remains of quantifying its causal impact, for which various measures such as transfer entropy or information flow have been proposed [14–18]. We compare how our work relates to these measures in Sec. 4.1.

The intersection of RL and causality has been the subject of recent research [19–23]. Close to ours is the work of Pitis et al. [24], who also use influence detection, albeit to create counterfactual data that augments the training of RL agents. In Sec. 5, we find that our approach to action influence detection performs better than their heuristic approach. Additionally, we demonstrate that influence detection can also be used to help agents explore better. To this end, we use influence as a type of intrinsic motivation. For exploration, various signals have been proposed, e.g. model surprise [25–27], learning progress [27, 28], empowerment [29, 30], information gain [31–33], or predictive information [34, 35]. Inspired by causality, Sontakke et al. [36] introduce an exploration signal that leads agents to experiment with the environment to discover causal factors of variation. In concurrent work, Zhao et al. [37] propose to use mutual information between the agent and the environment state for exploration. As in our work, the agent is considered a separate entity from the environment. However, their approach does not discriminate between individual situations the agent is in. Causal influence is also related to the concept of contingency awareness from psychology [38], that is, the knowledge that one's actions can affect the environment. On Atari games, exploring through the lens of contingency awareness has led to state-of-the-art results [39, 40].

3 Background

We are concerned with a Markov decision process $\langle \mathcal{S}, \mathcal{A}, P, r, \gamma \rangle$ consisting of state and action space, transition distribution, reward function and discount factor.[1] Most real world environments consist of entities that behave mostly independently of each other. We model this by assuming a known state space factorization $\mathcal{S} = \mathcal{S}_1 \times \ldots \times \mathcal{S}_N$, where each $\mathcal{S}_i$ corresponds to the state of an entity.

[1] We use capital letters (e.g. X) to denote random variables, small letters to denote samples drawn from particular distributions (e.g. $x \sim P_X$), and caligraphy letters to denote graphs, sets and sample spaces (e.g. $x \in \mathcal{X}$). We denote distributions with P and their densitites with p.



(a) Causal Graph $\mathcal{G}$

(b) No influence of A on S'_2

(c) Influence of A on S'_1 and S'_2

Figure 1: Causal graphical model capturing the environment transition from state S to S' by action A, factorized into state components. (a): Viewed globally over all time steps, all components of the state and the action can influence all state components at the next time step. (b, c): Given a situation $S = s$, some influences may or may not not hold in the *local causal graph* $\mathcal{G}_{S=s}$. In this paper, our aim is to detect which influence the action has on S', i.e. the presence of the red arrows.

3.1 Causal Graphical Models

We can model the one-step transition dynamics at time step t using a *causal graphical model* (CGM) [3, 10] over the set of random variables $\mathcal{V} = \{S_1, \ldots, S_N, A, S'_1, \ldots, S'_N\}$, consisting of a directed graph $\mathcal{G}$ (see Fig. 1a) and a conditional distribution $P(V_i \mid \mathrm{Pa}(V_i))$ for each node $V_i \in \mathcal{V}$, where $\mathrm{Pa}(V_i)$ is the set of parents of V_i in the causal graph. We assume that the joint distribution $P_{\mathcal{V}}$ is Markovian with respect to the graph [3, Def. 6.21 (iii)], that is, its density exists and factorizes as

$$p(v_1, \ldots, v_{|\mathcal{V}|}) = \prod_{i=1}^{|\mathcal{V}|} p(v_i \mid \mathrm{Pa}(V_i)). \tag{1}$$

In a CGM, we can model a (stochastic) intervention $\mathrm{do}(V_i := q(v_i \mid \mathrm{Pa}(V_i)))$ on variable V_i by replacing its conditional $p(v_i \mid \mathrm{Pa}(V_i))$ in Eq. 1 with the distribution $q(v_i \mid \mathrm{Pa}(V_i))$ [3]. Here, V_i could e.g. be a state component S_i, or the agent's action A. Thus, whereas a probabilistic graphical model represents a single distribution, a CGM represents a set of distributions [4].

The causal graph that we assume is shown in Fig. 1a. Within a time step, there are no edges, i.e. no instantaneous effects, except for the action which is computed by the policy $\pi(A \mid S)$. Between time steps, the graph is fully connected. The reason is that whenever an interaction between two components S_i and S_j, however unlikely, is possible, it is necessary to include an arrow $S_i \to S'_j$ (and vice versa). Nevertheless, during most concrete time steps, there should be *no* interaction between entities, reflecting the assumption that the state components represent independent entities in the world. In particular, the agent's "sphere of influence" (depicted in blue in Figs. 1b and 1c) is limited – its action A can only sparsely affect other entities. Thus, in this paper, we are interested in inferring the influence the action has in a specific state configuration $S = s$, that is, the *local causal model* in s.

Definition 1. Given a CGM with distribution $P_{\mathcal{V}}$ and graph $\mathcal{G}$, we define the *local CGM* induced by observing $X = x$ with $X \subset \mathcal{V}$ to be the CGM with joint distribution $P_{\mathcal{V}|X=x}$ and the graph $\mathcal{G}_{X=x}$ resulting from removing edges from $\mathcal{G}$ until $P_{\mathcal{V}|X=x}$ is causally minimal with respect to the graph.

Causal minimality tells us that each edge $X \to Y$ in the graph must be "active", in the sense that Y is conditionally dependent on X given all other parents of Y [3, Prop. 6.36]. In the following, we assume that $P_{\mathcal{V}}$ satifies causal minimality with respect to $\mathcal{G}$. In particular, this precludes that the action A is chosen deterministically by the policy for any $s \in \mathcal{S}$, which could "hide" the causal influence of the agent in s.

3.2 The Cause of an Effect

When is an agent's action $A = a$ the cause of an outcome $B = b$? Answering this question precisely is surprisingly non-trivial and is studied under the name of *actual causation* [10, 41]. Humans would answer by contrasting the actual outcome to some normative world in which $A = a$ did not happen,

i.e. they would ask the counterfactual question "What would have happened normally to B without $A = a$?" [41]. Algorithmitizing this approach poses certain problems. First, it requires a "normal" outcome which can be difficult to compute as it depends on the behavior of the different actors in the world. Second, it requires to actually observe the world's state without the agents interference. Such a "no influence" action may not be available for every agent. Instead, we are inspired by an alternative approach, the so-called *"but-for"* test: "$B = b$ would not have happened but for $A = a$." In other words, $A = a$ was a necessary condition for $B = b$ to occur, and under a different value for A, B would have had a different value as well. This matches well with an algorithmic view on causation: A is a cause of B if the value of A is required to determine the value of B [42].

The but-for test yields potentially counterintuitive assessments. Consider a robotic arm close to an object but performing an action that moves it away from the object. Then this action is considered a cause for the position of the object in this step, as an alternative action touching the object would have led to a different outcome. Algorithmically, knowing the action is required to determine what happens to the object – all actions are considered to be a cause in this situation. Importantly, this implies that we cannot differentiate whether individual actions are causes or not, but can only identify whether or not the agent has causal influence on other entities in the current state.

4 Causal Influence Detection

As the previous discussion showed, having causal influence is dependent on the situation the agent is in, rather than the chosen actions. We characterize this as the agent being *in control*, analogous to similar notions in control theory [43]. Formally, using the causal model introduced in Sec. 3, *we define the agent to be in control of S'_j in state $S = s$ if there is an edge $A \to S'_j$ in the local causal graph $\mathcal{G}_{S=s}$*. The following proposition characterizes when this is the case (proofs in Suppl. A.1).

Proposition 1. *Let $\mathcal{G}_{S=s}$ be the graph of the local CGM induced by $S = s$. There is an edge $A \to S'_j$ in $\mathcal{G}_{S=s}$ if and only if $S'_j \not\perp\!\!\!\perp A \mid S = s$ for some $j \in \{1, \dots, n\}$. Further, this holds under all possible interventions $\mathrm{do}(A := \pi(a|s))$ with π having full support.*

To detect when the agent is in control, we can intervene with a policy. The following proposition gives conditions under which conclusions drawn from one policy generalize to many policies.

Proposition 2. *If there is an intervention $\mathrm{do}(A := \pi(a|s))$ with π having full support for which it holds that $S'_j \perp\!\!\!\perp A \mid S = s$, this independence holds under* all *possible interventions and the agent is not in control in s. If there is an intervention $\mathrm{do}(A := \pi(a|s))$ for which it holds that $S'_j \not\perp\!\!\!\perp A \mid S = s$, this dependence holds under* all *interventions with full support, and the agent is in control in s.*

4.1 Measuring Causal Action Influence

Our goal is to find a state-dependent quantity that measures whether the agent is in control of S'_j. As Prop. 1 tells us, control (or its absence) is linked to the independence $S'_j \perp\!\!\!\perp A \mid S = s$. A well-known measure of dependence is the conditional mutual information (CMI) [44] which is zero for independence. We thus propose to use (pointwise) CMI as a measure of *causal action influence* (CAI) that can be thresholded to get a classification of control (see Suppl. A.2 for a derivation):

$$C^j(s) := I(S'_j; A \mid S = s) = \mathbb{E}_{a \sim \pi}\Big[\mathrm{D_{KL}}\Big(P_{S'_j|s,a} \,\Big\|\, P_{S'_j|s}\Big)\Big]. \tag{2}$$

We want this measure to be independent of the particular policy used in the joint distribution $P(S, A, S')$. This is because we might not be able to sample from or evaluate this policy (e.g. in off-policy RL, the data stems from a mixture of different policies). Fortunately, Prop. 2 shows that to detect control, it is sufficient to demonstrate (in-)dependence for a single policy with full support. Thus, we can choose a uniform distribution over the action space as the policy: $\pi(A) = \mathcal{U}(\mathcal{A})$.

Let us discuss how CAI relates to previously suggested measures of (causal) influence. *Transfer entropy* [14] is a non-linear extension of Granger causality [45] quantifying causal influence in time series under certain conditions. CAI is similar to a one-step, local transfer entropy [17] with the difference that CAI conditions on the full state S. Janzing et al. [18] put forward a measure of *causal strength* fulfilling several natural criteria that other measures, including transfer entropy, fail to satisfy. In Suppl. A.3, we show that CAI is a pointwise version of Janzing et al.'s causal strength, for policies not conditional on the state S (adding further justification for the choice of a uniform random

policy). Furthermore, we can relate CAI to notions of *controllability* [43]. Decomposing $C^j(s)$ as $H(S'_j \mid s) - H(S'_j \mid A, s)$, where H denotes the conditional entropy [44], we can interpret CAI as quantifiying the degree to which S'_j can be controlled in s, accounting for the system's intrinsic uncertainty that cannot be reduced by the action.

In the context of RL, *empowerment* [29, 30, 46] is a well-known quantity used for intrinsically-motivated exploration that leads agents to states of maximal influence over the environment. Empowerment, for a state s, is defined as the channel capacity between action and a future state, which coincides with $\max_\pi C(s)$ for one-step empowerment. CAI can thus be seen as a non-trivial lower bound of empowerment that is easier to compute. However, CAI differs from empowerment in that it does not treat the state space as monolithic and is specific to an entity. In Sec. 6.1, we demonstrate that an RL agent maximizing CAI quickly achieves control over its environment.

4.2 Learning to Detect Control

Estimating CMI is a hard problem on many levels: it involves computing high dimensional integrals, representing complicated distributions and having access to limited data; strictly speaking, each conditioning point s is seen only once in continuous spaces. In practice, one thus has to resort to an approximation. Non-parametric estimators based on nearest neighbors [47, 48] or kernels methods [49] are known to not scale well to higher dimensions [50]. Instead, we approach the problem by learning neural network models with suitable simplifying assumptions.

Expanding the KL divergence in Eq. 2, we can write CAI as

$$C^j(s) = I(S'_j; A \mid S = s) = \mathbb{E}_A \mathbb{E}_{S'_j \mid s,a} \left[\log \frac{p(s'_j \mid s, a)}{\int p(s'_j \mid s, a)\, \pi(s) \mathrm{d}a} \right] \tag{3}$$

To compute this term, we estimate the transition distribution $p(s'_j \mid s, a)$ from data. We then approximate the outer expectation and the transition marginal $p(s'_j \mid s)$ by sampling K actions from the policy π. This gives us the estimator

$$\hat{C}^j(s) = \frac{1}{K} \sum_{i=1}^{K} \left[\mathrm{D_{KL}} \left(p(s'_j \mid s, a^{(i)}) \;\middle\|\; \frac{1}{K} \sum_{k=1}^{K} p(s'_j \mid s, a^{(k)}) \right) \right], \tag{4}$$

with $\{a^{(1)}, \ldots, a^{(K)}\} \overset{\text{iid}}{\sim} \pi$. Here, we replaced the infinite mixture $p(s'_j \mid s)$ with a finite mixture, $p(s'_j \mid s) \approx \frac{1}{K} \sum_{i=1}^{K} p\big(s'_j \mid s, a^{(i)}\big)$, and used Monte-Carlo to approximate the expectation. Poole et al. [51] show that this estimator is a lower bound converging to the true mutual information $I(S'_j; A \mid S = s)$ as K increases (assuming, however, the true density $p(s'_j \mid s, a)$).

To compute the KL divergence itself, we make the simplifying assumption that the transition distribution $p(s'_j \mid s, a)$ is normally distributed given the action, which is reasonable in the robotics environment we are targeting. This allows us to estimate the KL without expensive MC sampling by using an approximation for mixtures of Gaussians from Durrieu et al. [52]. We detail the exact formula we use in Suppl. A.4.

With the normality assumption, the density itself can be learned using a probabilistic neural network and simple maximum likelihood estimation. That is, we parametrize $p(s'_j \mid s, a)$ as $\mathcal{N}(s'_j; \mu_\theta(s, a), \sigma^2_\theta(s, a))$, where $\mu_\theta, \sigma^2_\theta$ are the outputs of a neural network $f_\theta(s, a)$. We find the parameters θ by minimizing the negative log-likelihood over samples $\mathcal{D} = \{(s^{(i)}, a^{(i)}, s'^{(i)})\}_{i=1}^N$ collected by some policy:

$$\theta^* = \arg\min_\theta \frac{1}{N} \sum_{i=1}^{N} \frac{\left\| s'^{(i)}_j - \mu_\theta\big(s^{(i)}, a^{(i)}\big) \right\|_2^2}{2\sigma^2_\theta\big(s^{(i)}, a^{(i)}\big)} + \log \sigma_\theta\big(s^{(i)}, a^{(i)}\big). \tag{5}$$

There are some intricacies regarding the policy that collects the data for model training and the sampling policy π that is used to compute CAI. First of all, the two policies need to have overlapping support to avoid evaluating the model under actions never seen during training. Furthermore, if the data policy is different from the sampling policy π, the model is biased to some degree. This suggests to use π for collecting the data; however, as we use a random policy, this will not result in interesting data in most environments. The bias can be reduced by sampling actions from π during data collection with some probability and only train on those. In practice, however, we find to obtain better performing models by training on all data despite potentially being biased.

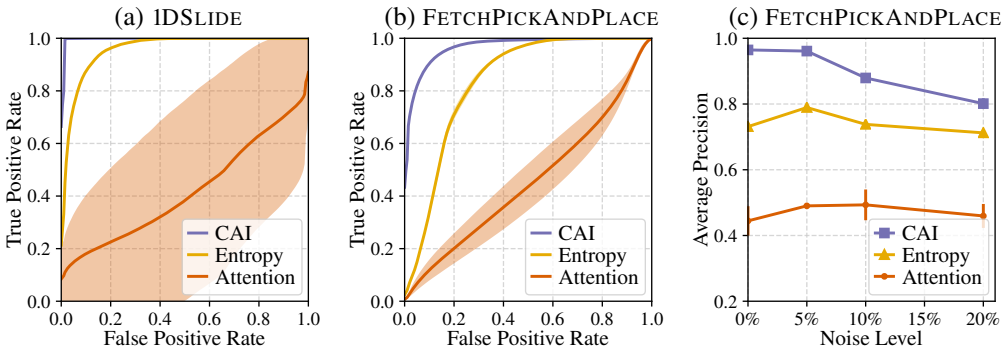


Figure 2: Causal influence detection performance. (a, b) ROC curves on 1DSLIDE and FETCHPICKANDPLACE environments. (c) Average precision for FETCHPICKANDPLACE depending on added state noise. Noise level is given as percentage of one standard deviation over the dataset.

Table 1: Results for evaluating causal influence detection on different environments. We measure area under the ROC curve (AUC), average precision (AP), and the best achievable F-score (F_1).

	1DSLIDE			FETCHPICKANDPLACE		
	AUC	AP	F_1	AUC	AP	F_1
CAI (ours)	1.00 ± 0.00	0.98 ± 0.00	0.95 ± 0.01	0.97 ± 0.01	0.96 ± 0.00	0.89 ± 0.00
Entropy	0.96 ± 0.00	0.47 ± 0.01	0.50 ± 0.01	0.84 ± 0.00	0.73 ± 0.00	0.78 ± 0.00
Attention [24]	0.42 ± 0.31	0.13 ± 0.14	0.18 ± 0.17	0.46 ± 0.06	0.44 ± 0.04	0.62 ± 0.00
Contacts	0.89	0.78	0.88	0.79	0.77	0.73

5 Empirical Evaluation of Causal Influence Detection

In this section, we evaluate the quality of our proposed causal influence detection approach in relevant environments. As a simple test case, we designed an environment (1DSLIDE) in which the agent must slide an object to a goal location by colliding with it. Furthermore, we test on the FETCHPICKANDPLACE environment from OpenAI Gym [53], in its original setting and when adding Gaussian noise to the observations to simulate more real-world conditions. In both environments, the target variables of interest are the coordinates of the object. Note that we need the true causal graph at each time step for the evaluation. For 1DSLIDE, we derive this information from the simulation. For the pick and place environment with its non-trivial dynamics, we resort to a heuristic of the possible movement range of the robotic arm in one step. Detailed information about the setup is provided in Suppls. B and E.

For our method, we use CAI estimated according to Eq. 4 (with $K = 64$) as a classification score that is thresholded to gain a binary decision. We compare with a recently proposed method [24] that uses the attention weights of a Transformer model [54] to model influence. Moreover, we compare with an *Entropy* baseline that uses $H(S'_j \mid s)$ as a score and a *Contact* baseline based on binary contact information from the simulator. We show the test results over 5 random seeds in Table 1 and Fig. 2. We observe that CAI is able to reliably detect causal influence and no other baseline is able to do so. When increasing the observation noise, the performance drops gracefully for CAI as shown in Fig. 2c. Suppl. C contains more experimental results, including a visualization of CAI's behavior.

6 Improving Efficiency in Reinforcement Learning

Having established the efficacy of our causal action influence (CAI) measure, we now develop several approaches to use it to improve RL algorithms.We will empirically verify the following claims in robotic manipulation environments: CAI improves sample efficiency and performance by (i) better state exploration through an exploration bonus, (ii) causal action exploration, and (iii) prioritizing experiences with causal influence during training.

We consider the environments FETCHPUSH, FETCHPICKANDPLACE from OpenAI Gym [55], and FETCHROTTABLE which is our modification containing a rotating table (explained in Suppl. B.3).

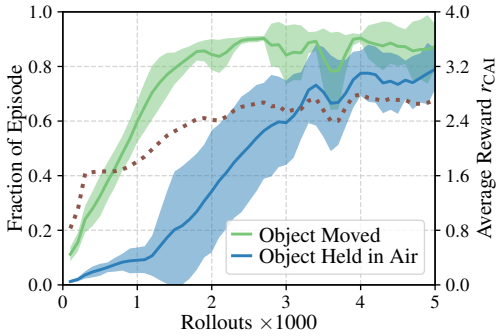


Figure 3: Intrinsically motivated learning on FETCHPICKANDPLACE. The reward is only r_{CAI} measured on the object coordinates.

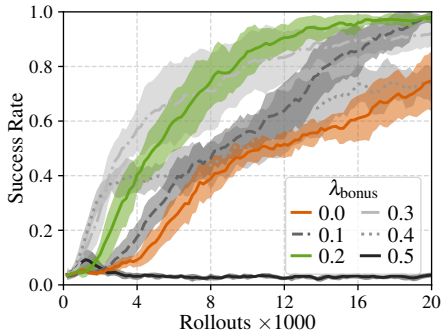


Figure 4: *Exploration bonus* improves performance in FETCHPICKANDPLACE. Sensitivity to the bonus reward scale λ_{bonus}.

These environments are goal-conditioned RL tasks with sparse rewards, meaning that each episode, a new goal is provided and the agent only receives a distinct reward upon reaching it. We use DDPG [56] with hindsight experience replay (HER) [57] as the base RL algorithm, a combination that achieves state-of-the-art results in these environment. The influence detection model is trained online on the data collected from an RL agent learning to solve its task. Since our measure C^j requires an entity of interest, we choose the coordinates of the object (as S_j). In all experiments, we report the mean success rate with standard deviation over 10 random seeds. More information about the experimental settings can be found in Suppl. F.

6.1 Intrinsic Motivation to Seek Influence

Causal Action Influence as Reward Bonus. We hypothesize that it is useful for an agent to be intrinsically motivated to gain control over its environment. We test this hypothesis by letting the agent maximize the causal influence it has over entities of interest. This can be achieved by using our influence measure as a reward signal. The reward signal can be used on its own, as an intrinsic motivation-type objective, or in conjunction with a task-specific reward as an exploration bonus. In the former case, we expect the agent to discover useful behaviors that can help it master task-oriented skills afterwards; in the latter case, we expect learning efficiency to improve, especially in sparse extrinsic reward scenarios. Concretely, for a state s, we define the bonus as $r_{\text{CAI}}(s) = C^j(s)$, and the total reward as $r(s) = r_{\text{task}}(s) + \lambda_{\text{bonus}}\, r_{\text{CAI}}(s)$, where $r_{\text{task}}(s)$ is the task reward, and λ_{bonus} is a hyperparameter.

Experiment on Intrinsically Motivated Learning. We first test the behavior of the agent in the absence of any task-specific reward on the FETCHPICKANDPLACE environment. Interestingly, the agent learned to grasp, lift, and hold the object in the air already after 2000 episodes, as shown in Fig. 3. The results demonstrate that encouraging causal control over the environment is well suited to prepare the agent for further tasks it might have to solve.

Impact of CAI Reward Bonus. Second, we are interested in the impact of adding an exploration bonus. In Fig. 4, we present results on the FETCHPICKANDPLACE environment when varying the reward scale λ_{bonus}. Naturally, the exploration bonus needs to be selected in the appropriate scale as a value too high will make it dominate the task reward. If selected correctly, the sample efficiency is improved drastically; for example, we find that the agent reaches a success rate of 60% four-times faster than the baseline (DDPG+HER) without any bonus ($\lambda_{\text{bonus}} = 0$).

6.2 Actively Exploring Actions with Causal Influence

Following Actions with the Most Causal Influence. Exploration via bonus rewards favors the re-visitation of already seen states. An alternative approach to exploration uses pro-active planning to choose exploratory actions. In our case, we can make use of our learned influence estimator to pick actions which we expect will have the largest causal effect on the agent's surroundings. From a causal viewpoint, the resulting agent can be seen as an experimenter that performs planned interventions in the environment to verify its beliefs. Should the actual outcome differ from the expected outcome, subsequent model updates can integrate the new data to self-correct the causal influence estimator.

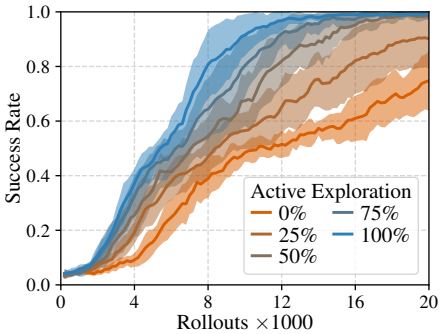


Figure 5: Performance of *active exploration* in FETCHPICKANDPLACE depending on the fraction of exploratory actions chosen actively (Eq. 6) from a total of 30% exploratory actions.

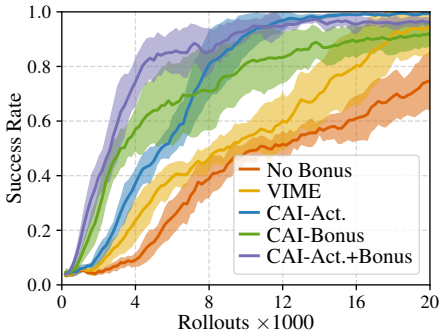


Figure 6: Experiment comparing exploration strategies on FETCHPICKANDPLACE. The combination of active exploration and reward bonus yields the largest sample efficiency.

Concretely, given the agent being in state s, we choose the action that has the largest contribution to the empirical mean in Eq. 4:

$$a^* = \underset{a \in \{a^{(1)}, \dots, a^{(K)}\}}{\arg\max} \mathrm{D}_{\mathrm{KL}} \left(p(s'_j \mid s, a) \,\Big\|\, \frac{1}{K} \sum_{i=1}^{K} p(s'_j \mid s, a^{(i)}) \right), \tag{6}$$

with $\{a^{(1)}, \dots, a^{(K)}\} \overset{\text{iid}}{\sim} \pi$. Intuitively, the selected action will be the one which results in maximal deviation from the expected outcome under all actions. For states s where the the agent is not in control, i.e. $C^j(s) \approx 0$, the action selection is uniform at random.

Active Exploration in Practice. Can active exploration replace ϵ-greedy exploration? To gain insights, we study the impact of the fraction of actively chosen exploration actions. For every exploratory action (ϵ is 30% in our experiments), we choose an action according to Eq. 6 the specified fraction of the time, and otherwise a random action. Figure 5 shows that any amount of active exploration improves over simple random exploration. Active causal action exploration can improve the sample efficiency roughly by a factor of two.

Combined CAI Exploration. We also present the combination of reward bonus and active exploration and compare our method with VIME, another exploration scheme based on information-theoretic measures [33]. In contrast to our method, VIME maximizes the information gain about the state transition dynamics (we compare different VIME variants in Suppl. D). Figure 6 quantifies the superiority of all CAI variants and shows that combining the two exploration strategies compounds to increase sample efficiency even further. In the figure, CAI uses 100% active exploration and $\lambda_{\text{bonus}} = 0.2$ as the bonus reward scale.

6.3 Causal Influence-based Experience Replay

Prioritizing According to Causal Influence. We will now propose another method using CAI, namely to inform the choice of samples replayed to the agent during off-policy training. Typically, past states are sampled uniformly for learning. Intuitively, focusing on those states where the agent has control over the object of interest (as measured by CAI) should improve the sample efficiency. We can implement this idea using a prioritization scheme that samples past episodes more frequently in which the agent had more influence. Concretely, we define the probability $P^{(i)}$ of sampling any state s in episode i (of M episodes) in the replay buffer as

$$P^{(i)} = \frac{p^{(i)}}{\sum_{i=1}^{M} p^{(i)}} \cdot \frac{1}{T}, \qquad \text{with} \qquad p^{(i)} = \left(M + 1 - \operatorname*{rank}_{\mathrm{i}} \sum_{t=1}^{T} C^j\left(s^{(t)}\right) \right)^{-1}. \tag{7, 8}$$

where $p^{(i)}$ is the priority of episode i and T is the episode length. The priority of an episode is based on the (inverse) rank of the episode when sorting all M episodes according to their total influence. We call this *causal action influence prioritization* (CAI-P).

This scheme is similar to Prioritized Experience Replay [58], with two differences: instead of using the TD error for prioritization, we use the causal influence measure. Furthermore, instead of

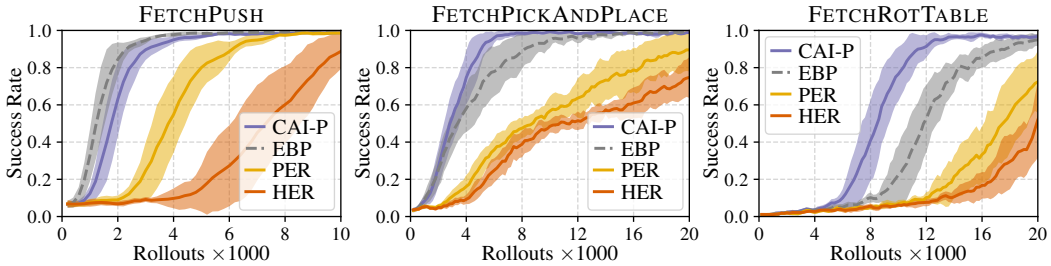


Figure 7: Prioritizing experience replay in different manipulation environments. Comparison of causal action influence prioritization (CAI-P) against baselines: the energy-based method (EBP) [59] with privileged information, prioritized experience replay (PER) [58], and HER without prioritization.

prioritizing individual states, we prioritize episodes and sample states uniformly within episodes. This is because the information about the return that can be achieved from an influence state still needs to be propagated back to non-influenced states by TD updates, which requires sampling them.

Influence-Based Prioritization in Manipulation Tasks. We compare our influence-based prioritization (CAI-P) against no prioritization in hindsight experience replay (HER) (a strong baseline for multi-goal RL), and two other prioritization schemes: prioritized experience replay (PER) [58] and energy-based prioritization (EBP) [59]. Especially EBP is a strong method for the environments we are considering as it uses privileged knowledge of the underlying physics to replay episodes based on the amount of energy that is transferred from agent to the object to manipulate. All prioritization variants are equipped with HER as well. The FETCHROTTABLE environment, as shown in Fig. 8, is an interesting test bed, as the objects can move by the rotating table without the control of the agent. The results, as shown in Fig. 7, show that causal influence prioritization can speed up learning drastically. Our method is on par or better than the energy-based (oracle) method EBP and improves over PER by a factor of 1.5–2.5 in learning speed (at 60% success rate). Finally, in Suppl. D, we combine all our proposed improvements and show that FETCHPICKANDPLACE can be solved up to 95% success rate in just 3000 episodes.

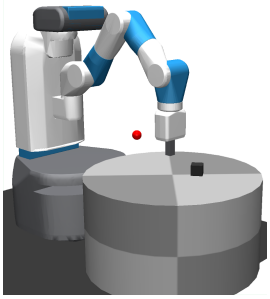


Figure 8: FETCHROTTABLE. The table rotates periodically.

7 Discussion

In this work, we show how situation-dependent causal influence detection can help improve reinforcement learning agents. To this end, we derive a measure of local causal action influence (CAI) and introduce a data-driven approach based on neural network models to estimate it. We showcase using CAI as an exploration bonus, as a way to perform active action exploration, and to prioritize in experience replay. Each of our applications yields strong improvements in sample efficiency. We expect that there are further ways to use our causal measure in RL, e.g. for credit assignment.

Our work has several limitations. First, we assume full observability of the state, which simplifies the causal inference problem as there is no confounding between an agent's action and its effect. Under partial observability, our approach could still be applicable using latent variable models [60]. Second, we require an available factorization of the state into causal variables. The problem of automatically learning causal variables from high-dimensional data is open [4] and our method would likely benefit from advances in this field. Third, the accurate estimation of our measure relies on a correct model. We found that deep networks can struggle at times to pick up the causal relationship between actions and entities. How to design models with appropriate inductive biases for cause-effect inference is an open question [3, 4, 61].

An intriguing future direction is to extend our work to influence detection between entities, a prerequisite for identifying multi-step influences of the agent on the environment. Being able to model such indirect interventions would bring us closer to "artificial scientists" – agents that can perform planned experiments to reveal the latent causal structure of the world.

References

[1] Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.

[2] Dmitry Kalashnikov, Alex Irpan, Peter Pastor Sampedro, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, and Sergey Levine. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on Robot Learning 2018*, 2018. URL https://arxiv.org/pdf/1806.10293.

[3] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press, Cambridge, MA, USA, 2017.

[4] B. Schölkopf, F. Locatello, S. Bauer, R. Nan Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio. Towards causal representation learning. *Proceedings of the IEEE*, 2021. doi: 10.1109/JPROC.2021.3058954.

[5] Giambattista Parascandolo, Niki Kilbertus, Mateo Rojas-Carulla, and Bernhard Schölkopf. Learning independent causal mechanisms. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4036–4044. PMLR, 10–15 Jul 2018. URL http://proceedings.mlr.press/v80/parascandolo18a.html.

[6] Yoshua Bengio, Tristan Deleu, Nasim Rahaman, Nan Rosemary Ke, Sebastien Lachapelle, Olexa Bilaniuk, Anirudh Goyal, and Christopher Pal. A meta-transfer objective for learning to disentangle causal mechanisms. In *8th International Conference on Learning Representations (ICLR)*, 2020. URL https://openreview.net/forum?id=ryxWIgBFPS.

[7] Francesco Locatello, Ben Poole, Gunnar Raetsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6348–6359. PMLR, 13–18 Jul 2020. URL http://proceedings.mlr.press/v119/locatello20a.html.

[8] Nan Rosemary Ke, Olexa Bilaniuk, Anirudh Goyal, Stefan Bauer, Hugo Larochelle, Bernhard Schölkopf, Michael C. Mozer, Chris Pal, and Yoshua Bengio. Learning neural causal models from unknown interventions. *ArXiv*, abs/1910.01075, 2019.

[9] A. Goyal, A. Lamb, J. Hoffmann, S. Sodhani, S. Levine, Y. Bengio, and B. Schölkopf. Recurrent independent mechanisms. In *9th International Conference on Learning Representations (ICLR)*, 2021.

[10] Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, USA, 2nd edition, 2009.

[11] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. MIT Press, Cambridge, MA, 2nd edition, 2000.

[12] J. Peters, J. Mooij, D. Janzing, and B. Schölkopf. Identifiability of causal graphs using functional models. In F. G. Cozman and A. Pfeffer, editors, *27th Conference on Uncertainty in Artificial Intelligence*, pages 589–598, Corvallis, OR, 2011. AUAI Press.

[13] Michael Eichler. *Causal Inference in Time Series Analysis*. Wiley, 2012. ISBN 9781119945710. doi: 10.1002/9781119945710.ch22.

[14] Thomas Schreiber. Measuring information transfer. *Physical Review Letters*, 85 2:461–4, 2000.

[15] Nihat Ay and David C. Krakauer. Geometric robustness theory and biological networks. *Theory in Biosciences*, 125(2):93–121, 2007. ISSN 1431-7613. doi: https://doi.org/10.1016/j.thbio.2006.06.002. URL https://www.sciencedirect.com/science/article/pii/S1431761306000255.

[16] N. Ay and D. Polani. Information flows in causal networks. *Adv. Complex Syst.*, 11:17–41, 2008.

[17] Joseph T. Lizier. *The local information dynamics of distributed computation in complex systems*. PhD thesis, University of Sydney, 2013. URL http://d-nb.info/1024629171.

[18] D. Janzing, D. Balduzzi, M. Grosse-Wentrup, and B. Schölkopf. Quantifying causal influences. *Annals of Statistics*, 41(5):2324–2358, 2013. URL http://projecteuclid.org/euclid.aos/1383661266.

[19] E. Bareinboim, A. Forney, and J. Pearl. Bandits with unobserved confounders: A causal approach. In *Advances in Neural Information Processing*, volume 28. Curran Associates, Inc., 2015.

[20] Chaochao Lu, B. Schölkopf, and José Miguel Hernández-Lobato. Deconfounding reinforcement learning in observational settings. *ArXiv*, abs/1812.10576, 2018.

[21] Danilo Jimenez Rezende, Ivo Danihelka, George Papamakarios, N. Ke, Ray Jiang, T. Weber, K. Gregor, Hamza Merzic, Fabio Viola, J. Wang, Jovana Mitrovic, F. Besse, Ioannis Antonoglou, and Lars Buesing. Causally Correct Partial Models for Reinforcement Learning. *ArXiv*, abs/2002.02836, 2020.

[22] Lars Buesing, T. Weber, Yori Zwols, Sébastien Racanière, A. Guez, J. Lespiau, and N. Heess. Woulda, Coulda, Shoulda: Counterfactually-Guided Policy Search. In *7th International Conference on Learning Representations (ICLR)*, 2019.

[23] J. Zhang and E. Bareinboim. Near-optimal reinforcement learning in dynamic treatment regimes. In *Advances in Neural Information Processing Systems*, volume 33. Curran Associates, Inc., 2019.

[24] Silviu Pitis, Elliot Creager, and Animesh Garg. Counterfactual data augmentation using locally factored dynamics. In *Advances in Neural Information Processing Systems*, volume 33. Curran Associates, Inc., 2020.

[25] J. Schmidhuber. Curious model-building control systems. In *Proceedings IEEE International Joint Conference on Neural Networks*, pages 1458–1463 vol.2, 1991. doi: 10.1109/IJCNN.1991.170605.

[26] Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 488–489, 2017.

[27] Sebastian Blaes, Marin Vlastelica, Jia-Jie Zhu, and Georg Martius. Control What You Can: Intrinsically motivated task-planning agent. In *Advances in Neural Information Processing*, pages 12520–12531. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/9418-control-what-you-can-intrinsically-motivated-task-planning-agent.pdf.

[28] Cédric Colas, Pierre-Yves Oudeyer, Olivier Sigaud, Pierre Fournier, and Mohamed Chetouani. CURIOUS: intrinsically motivated modular multi-goal reinforcement learning. In *International Conference on Machine Learning (ICML'19)*, pages 1331–1340, 2019.

[29] A.S. Klyubin, D. Polani, and C.L. Nehaniv. Empowerment: a universal agent-centric measure of control. In *2005 IEEE Congress on Evolutionary Computation*, volume 1, pages 128–135 Vol.1, 2005. doi: 10.1109/CEC.2005.1554676.

[30] Shakir Mohamed and Danilo Jimenez Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper/2015/file/e00406144c1e7e35240afed70f34166a-Paper.pdf.

[31] Jan Storck, Sepp Hochreiter, and Jürgen Schmidhuber. Reinforcement driven information acquisition in non-deterministic environments. In *ICANN'95*, pages 159–164, 1995.

[32] Daniel Little and Friedrich Sommer. Learning and exploration in action-perception loops. *Frontiers in Neural Circuits*, 7:37, 2013. ISSN 1662-5110. doi: 10.3389/fncir.2013.00037. URL https://www.frontiersin.org/article/10.3389/fncir.2013.00037.

[33] Rein Houthooft, Xi Chen, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/abd815286ba1007abfbb8415b83ae2cf-Paper.pdf.

[34] Georg Martius, Ralf Der, and Nihat Ay. Information driven self-organization of complex robotic behaviors. *PLoS ONE*, 8(5):e63400, 2013. doi: 10.1371/journal.pone.0063400. URL http://dx.doi.org/10.1371/journal.pone.0063400.

[35] Keyan Zahedi, Georg Martius, and Nihat Ay. Linear combination of one-step predictive information with an external reward in an episodic policy gradient setting: a critical analysis. *Frontiers in Psychology*, 4(801), 2013. ISSN 1664-1078. doi: 10.3389/fpsyg.2013.00801. URL http://www.frontiersin.org/cognitive_science/10.3389/fpsyg.2013.00801/abstract.

[36] Sumedh A. Sontakke, Arash Mehrjou, Laurent Itti, and Bernhard Schölkopf. Causal Curiosity: RL agents discovering self-supervised experiments for causal representation learning. *ArXiv*, 2020. arXiv:2010.03110.

[37] Rui Zhao, Yang Gao, Pieter Abbeel, Volker Tresp, and Wei Xu. Mutual information state intrinsic control. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=0thEq8I5v1.

[38] John S. Watson. The development and generalization of contingency awareness in early infancy: Some hypotheses. *Merrill-Palmer Quarterly of Behavior and Development*, 12(2):123–135, 1966. ISSN 00260150. URL http://www.jstor.org/stable/23082793.

[39] Jongwook Choi, Yijie Guo, Marcin Moczulski, Junhyuk Oh, Neal Wu, Mohammad Norouzi, and Honglak Lee. Contingency-aware exploration in reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2019. URL https://openreview.net/forum?id=HyxGB2AcY7.

[40] Yuhang Song, Jianyi Wang, Thomas Lukasiewicz, Zhenghua Xu, Shangtong Zhang, Andrzej Wojcicki, and Mai Xu. Mega-Reward: Achieving human-level play without extrinsic rewards. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):5826–5833, Apr. 2020. doi: 10.1609/aaai.v34i04.6040. URL https://ojs.aaai.org/index.php/AAAI/article/view/6040.

[41] Joseph Y. Halpern. *Actual Causality*. The MIT Press, 2016. ISBN 9780262035026.

[42] J. Pearl, M. Glymour, and N. Jewell. *Causal Inference in Statistics: A Primer*. Wiley, 2016. ISBN 978-1-119-18684-7.

[43] Hugo Touchette and Seth Lloyd. Information-theoretic approach to the study of control systems. *Physica A: Statistical Mechanics and its Applications*, 331(1):140–172, 2004. ISSN 0378-4371. doi: https://doi.org/10.1016/j.physa.2003.09.007. URL https://www.sciencedirect.com/science/article/pii/S0378437103008100.

[44] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications and Signal Processing. Wiley-Interscience, 2nd ed edition, 2006.

[45] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–438, 1969. URL http://www.jstor.org/stable/1912791.

[46] Christoph Salge, Cornelius Glackin, and Daniel Polani. *Empowerment–An Introduction*, pages 67–114. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. ISBN 978-3-642-53734-9. doi: 10.1007/978-3-642-53734-9_4. URL https://doi.org/10.1007/978-3-642-53734-9_4.

[47] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 69:066138, 07 2004. doi: 10.1103/PhysRevE.69.066138.

[48] Jakob Runge. Conditional independence testing based on a nearest-neighbor estimator of conditional mutual information. In Amos Storkey and Fernando Perez-Cruz, editors, *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pages 938–947. PMLR, 09–11 Apr 2018. URL http://proceedings.mlr.press/v84/runge18a.html.

[49] Kirthevasan Kandasamy, Akshay Krishnamurthy, Barnabas Poczos, Larry Wasserman, and james m robins. Nonparametric von Mises estimators for entropies, divergences and mutual informations. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper/2015/file/06138bc5af6023646ede0e1f7c1eac75-Paper.pdf.

[50] Shuyang Gao, Greg Ver Steeg, and Aram Galstyan. Efficient Estimation of Mutual Information for Strongly Dependent Variables. In Guy Lebanon and S. V. N. Vishwanathan, editors, *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pages 277–286, San Diego, California, USA, 09–12 May 2015. PMLR. URL http://proceedings.mlr.press/v38/gao15.html.

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

[52] Jean-Louis Durrieu, J. Thiran, and Finnian Kelly. Lower and upper bounds for approximation of the kullback-leibler divergence between gaussian mixture models. *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4833–4836, 2012.

[53] Greg Brockman, Vicki Cheung, Ludwig Pettersson, J. Schneider, John Schulman, Jie Tang, and W. Zaremba. OpenAI Gym. *ArXiv*, abs/1606.01540, 2016.

[54] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[55] Matthias Plappert, Marcin Andrychowicz, Alex Ray, Bob McGrew, Bowen Baker, Glenn Powell, J. Schneider, Joshua Tobin, Maciek Chociej, P. Welinder, V. Kumar, and W. Zaremba. Multi-goal reinforcement learning: Challenging robotics environments and request for research. *ArXiv*, abs/1802.09464, 2018.

[56] T. Lillicrap, Jonathan J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2016.

[57] Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[58] Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. In *International Conference on Learning Representations (ICLR)*, 2016.

[59] Rui Zhao and Volker Tresp. Energy-based hindsight experience prioritization. In Aude Billard, Anca Dragan, Jan Peters, and Jun Morimoto, editors, *Proceedings of The 2nd Conference on Robot Learning*, volume 87 of *Proceedings of Machine Learning Research*, pages 113–122. PMLR, 29–31 Oct 2018.

[60] Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/94b5bde6de888ddf9cde6748ad2523d1-Paper.pdf.

[61] Anirudh Goyal and Yoshua Bengio. Inductive biases for deep learning of higher-level cognition. *ArXiv*, abs/2011.15091, 2020.

[62] E. Todorov, T. Erez, and Y. Tassa. MuJoCo: A physics engine for model-based control. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012.

[63] John R. Hershey and Peder A. Olsen. Approximating the Kullback Leibler divergence between gaussian mixture models. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, volume 4, pages IV–317–IV–320, 2007. doi: 10.1109/ICASSP.2007.366913.

[64] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

[65] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, 2010.

[66] Andrew M. Saxe, James L. McClelland, and S. Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.

[67] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2018.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] We discuss the limitations of our work in Sec. 7.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] Our work introduces a method for improving efficiency in reinforcement learning. We believe that there is no negative societal impact specific to our work that goes beyond the general concerns associated with deploying RL systems in the real world (such as issues of safety and underspecified targets).
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] All proofs are given in Suppl. A.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Code is provided under https://sites.google.com/view/cid-in-rl and will be made open source upon acceptance.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Training details are given in Sec. E and Sec. F.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] In Sec. E and Sec. F. Computing was conducted on an internal cluster.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes] We use environments from OpenAI Gym [53] based on the MuJoCo simulator [62]. We also adapted the code release of several baselines. We point to their URLs in Sec. F.

(b) Did you mention the license of the assets? [Yes] OpenAI Gym is released under MIT license. All baselines were also released under MIT license. MuJoCo is commercial software.

(c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Our newly introduced environments, 1DSLIDE and FETCHROTTABLE, are part of the code release.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]