A Proofs

Lemma 1. *Let $\mathbf{\Phi}^{\pi_1 *}$ be the set of all possible π-minimal state representations under π_1, where every $\Phi^{\pi_1 *} \in \mathbf{\Phi}^{\pi_1 *}$ is defined as $\Phi^{\pi_1 *} : \mathcal{S}^{\pi_1} \to \bar{\mathcal{S}}^{\pi_1 *}$ and $\bar{\mathcal{S}}^{\pi_1 *} = \times_i \operatorname{dom}(\bar{\Theta}^{\pi_1 * i})$, and let π_2 be a second policy such that for all $s_t \in \mathcal{S}^{\pi_1} \cap \mathcal{S}^{\pi_2}$,*

$$\operatorname{supp}\left(\pi_2(\cdot \mid s_t)\right) \subseteq \operatorname{supp}\left(\pi_1(\cdot \mid s_t)\right).$$

For all $\Phi^{\pi_1 *} \in \mathbf{\Phi}^{\pi_1 *}$, there exists a π-Markov state representation under policy π_2, $\Phi^{\pi_2} : \mathcal{S}^{\pi_2} \to \bar{\mathcal{S}}^{\pi_2}$ with $\bar{\mathcal{S}}^{\pi_2} = \times_i \operatorname{dom}(\bar{\Theta}^{\pi_2 i})$, such that $\bar{\Theta}^{\pi_2} \subseteq \bar{\Theta}^{\pi_1 *}$ for all $s_t \in \mathcal{S}^{\pi_1} \cap \mathcal{S}^{\pi_2}$. Moreover, there exist cases where $\bar{\Theta}^{\pi_2}_t \neq \bar{\Theta}^{\pi_1 *}_t$.

Proof. First, it is easy to show that

$$\forall s_t \in \mathcal{S}, \operatorname{supp}\left(\pi_2(\cdot \mid s_t)\right) \subseteq \operatorname{supp}\left(\pi_1(\cdot \mid s_t)\right) \iff \mathcal{S}^{\pi_2} \subseteq \mathcal{S}^{\pi_1},$$

and

$$\forall s_t \in \mathcal{S}, \operatorname{supp}\left(\pi_2(\cdot \mid s_t)\right) = \operatorname{supp}\left(\pi_1(\cdot \mid s_t)\right) \iff \mathcal{S}^{\pi_2} = \mathcal{S}^{\pi_1}.$$

In particular, $\mathcal{S}^{\pi_2} \subset \mathcal{S}^{\pi_1}$ if there is at least one state $s'_t \in \mathcal{S}^{\pi_1} \cap \mathcal{S}^{\pi_2}$ such that

$$\operatorname{supp}\left(\pi_2(\cdot \mid s'_t)\right) \subset \operatorname{supp}\left(\pi_1(\cdot \mid s'_t)\right)$$

while

$$\operatorname{supp}\left(\pi_2(\cdot \mid s_t)\right) = \operatorname{supp}\left(\pi_1(\cdot \mid s_t)\right)$$

for all other $s_t \in \mathcal{S}^{\pi_1} \cap \mathcal{S}^{\pi_2}$.

In such cases, we know that there is at least one action a' for which $\pi_2(a'_t \mid s'_t) = 0$ but $\pi_1(a'_t \mid s'_t) \neq 0$. Hence, if there was a state (or group of states) that could only be reached by taking action a'_t at s'_t, π_2 would never visit it and thus $\mathcal{S}^{\pi_2} \subset \mathcal{S}^{\pi_1}$.

Further, if $\mathcal{S}^{\pi_2} \subset \mathcal{S}^{\pi_1}$, we know that, for every $\Phi^{\pi_1 *} \in \mathbf{\Phi}^{\pi_1 *}$, there must be a $\Phi^{\pi_2 *}$ that requires, at most, the same number of variables, $\bar{\Theta}^{\pi_2}_t \subseteq \bar{\Theta}^{\pi_1 *}_t$ and, in some cases, fewer, $\bar{\Theta}^{\pi_1 *}_t \neq \bar{\Theta}^{\pi_2 *}_t$ (e.g., Frozen T-Maze example).

□

Proposition 1. *Let $\mathbf{\Phi}^*$ be the set of all possible minimal state representations, where every $\Phi^* \in \mathbf{\Phi}^*$ is defined as $\Phi^* : \mathcal{S} \to \bar{\mathcal{S}}^*$ with $\bar{\mathcal{S}}^* = \times_i \operatorname{dom}(\bar{\Theta}^{* i})$. For all π and all $\Phi^* \in \mathbf{\Phi}^*$, there exists a π-Markov state representation $\Phi^\pi : \mathcal{S}^\pi \to \bar{\mathcal{S}}^\pi$ with $\bar{\mathcal{S}}^\pi = \times_i \operatorname{dom}(\bar{\Theta}^{\pi i})$ such that for all $s \in \mathcal{S}^\pi$, $\bar{\Theta}^\pi \subseteq \bar{\Theta}^*$. Moreover, there exist cases for which $\bar{\Theta}^\pi$ is a proper subset, $\bar{\Theta}^\pi \neq \bar{\Theta}^*$.*

Proof. The proof follows from Lemma 1. We know that, in general, $\mathcal{S}^\pi \subseteq \mathcal{S}$, and if $\pi(a'_t | s'_t) = 0$ for at least one pair $a'_t \in \mathcal{A}$, $s'_t \in \mathcal{S}$ for which there is a state (or group of states) that can only be reached by taking action a'_t at s'_t, then $\mathcal{S}^\pi \subset \mathcal{S}$. Hence, for every Φ^* there is a Φ^π such that $\bar{\Theta}^\pi \subseteq \bar{\Theta}^*$, and in some cases, we may have $\bar{\Theta}^\pi \neq \bar{\Theta}^*$ (e.g., Frozen T-Maze example).

□

Theorem 1. *Let $\Phi^* : \mathcal{S} \to \bar{\mathcal{S}}^*$ with $\bar{\mathcal{S}}^* = \times_i \operatorname{dom}(\bar{\Theta}^{* i})$ be a minimal state representation. If, for some π, there is a π-Markov state representation $\Phi^\pi : \mathcal{S}^\pi \to \bar{\mathcal{S}}^\pi$ with $\bar{\mathcal{S}}^\pi = \times_i \operatorname{dom}(\bar{\Theta}^{\pi i})$, such that $\bar{\Theta}^\pi \subset \bar{\Theta}^*$ for some $s \in \mathcal{S}$, then Φ^π is confounded by policy π.*

Proof. Proof by contradiction. Let us assume that $\bar{\Theta}^\pi \subset \bar{\Theta}^*$, and yet there is no policy confounding. I.e., for all $s_t, s_{t+1} \in \mathcal{S}$, $a_t \in \mathcal{A}$,

$$R^\pi(\Phi^\pi(s_t), a_t) = R^\pi(\operatorname{do}(\Phi^\pi(s_t)), a_t) \tag{1}$$

and

$$\Pr^\pi(\Phi^\pi(s_{t+1}) \mid \Phi^\pi(s_t), a_t) = \Pr^\pi(\Phi^\pi(s_{t+1}) \mid \operatorname{do}(\Phi^\pi(s_t)), a_t) \tag{2}$$

First, note that the do-operator implies that the equality must hold for *all* s'_t in the equivalence of s_t class under Φ^π, $s'_t \in \{s_t\}^{\Phi^\pi} = \{h'_t \in H_t : \Phi(h'_t) = \Phi(h_t)\}$, i.e., not just those h'_t that are visited under π,

$$R^\pi(\Phi^\pi(s_t), a_t) = R^\pi(\operatorname{do}(\Phi^\pi(s_t)), a_t) = \{R(s'_t, a_t)\}_{s'_t \in \{s_t\}^\Phi} \tag{3}$$

which is precisely the first condition in Definition 4,

$$R(\Phi^\pi(s_t), a_t) = R(s_t, a_t), \tag{4}$$

for all $s_t \in \mathcal{S}$ and $a_t \in \mathcal{A}$.

Analogously, we have that,

$$\begin{aligned}\Pr^\pi(\Phi^\pi(s_{t+1}) \mid \Phi^\pi(s_t), a_t) &= \Pr^\pi(\Phi^\pi(s_{t+1}) \mid \mathrm{do}(\Phi^\pi(s_t)), a_t) \\ &= \Pr(\Phi^\pi(s_{t+1}) \mid \Phi^\pi(s_t), a_t)\end{aligned} \tag{5}$$

where the second equality reflects that the above must hold independently of π. Hence, we have that for all $s_t, s_{t+1} \in \mathcal{S}$ and $s'_t \in \{s_t\}^\Phi$,

$$\Pr(\Phi^\pi(s_{t+1}) \mid \Phi^\pi(s_t), a_t) = \Pr(\Phi^\pi(s_{t+1}) \mid \Phi^\pi(s'_t), a_t), \tag{6}$$

which means that, for all $s_t, s_{t+1} \in \mathcal{S}$ and $s_t \in \mathcal{A}$,

$$\begin{aligned}\Pr(\Phi^\pi(s_{t+1}) \mid \Phi^\pi(s_t), a_t) &= \Pr(\Phi^\pi(s_{t+1}) \mid s_t, a_t) \\ &= \sum_{s'_{t+1} \in \{s_{t+1}\}^{\Phi^\pi}} T(s'_{t+1} \mid s_t, a_t),\end{aligned} \tag{7}$$

which is the second condition in Definition 4.

Equations equation 4 and equation 7 reveal that if the assumption is true (i.e., Φ^π is not confounded by the policy), then Φ^π is not just π-Markov but actually strictly Markov (Definition 4). However, we know that $\Phi^*(s_t)$ is the minimal state representation, which contradicts the above statement, since, according to Definition 5, there is no proper subset of $\bar{\Theta}^*$, for all $s_t \in \mathcal{S}$, such that the representation remains Markov. Hence, $\bar{\Theta}^\pi \subset \bar{\Theta}^*$ implies policy confounding. □

Proposition 2. *Let $\{\bar{\Theta}^*\}_{\cup\Phi^*}$ be the union of variables in all possible minimal state representations. There exist cases where, for some π, there is a π-minimal state representation $\Phi^{\pi*} : \mathcal{S}^\pi \to \bar{\mathcal{S}}^{\pi*}$ with $\bar{\mathcal{S}}^{\pi*} = \times_i \operatorname{dom}(\bar{\Theta}^{\pi*i})$ such that $\bar{\Theta}^{\pi*} \setminus \{\bar{\Theta}^*\}_{\cup\Phi^*} \neq \emptyset$.*

Proof (sketch). Consider a deterministic MDP with a deterministic policy. Imagine there exists a variable X that is perfectly correlated with the episode's timestep t, but that is generally irrelevant to the task. The variable X would constitute in itself a valid π-Markov state representation since it can be used to determine transitions and rewards so long as a deterministic policy is followed. At the same time, X would not enter the minimal Markov state representation because it is useless under stochastic policies. Example 4 below illustrates this situation. □

B Example: Watch the Time

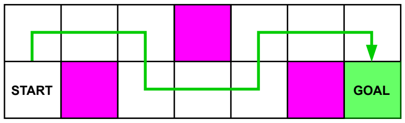

Figure 7: An illustration of the watch-the-time environment.

Example 4. (Watch the Time) This example is inspired by the empirical results of Song et al. (2020). Figure 7 shows a grid world environment., The agent must go from the start cell to the goal cell. The agent must avoid the pink cells; stepping on those yields a -0.1 penalty. There is a is $+1$ reward for reaching the goal. The agent can observe its own location within the maze X and the current timestep t. The two diagrams in Figure 8 are DBNs describing the environment dynamics. When actions are considered exogenous random variables (left diagram), the only way to estimate the reward at $t = 10$ is by looking at the agent's location. In contrast, when actions are determined by the policy (right diagram), t becomes a proxy for the agent's location X_{10}. This is because the start location and the sequence of actions are fixed. This implies that t is a perfectly valid π-Markov state representation under π^*. Moreover, as shown by the DBN on the right, the optimal policy may simply rely on t to determine the optimal action.

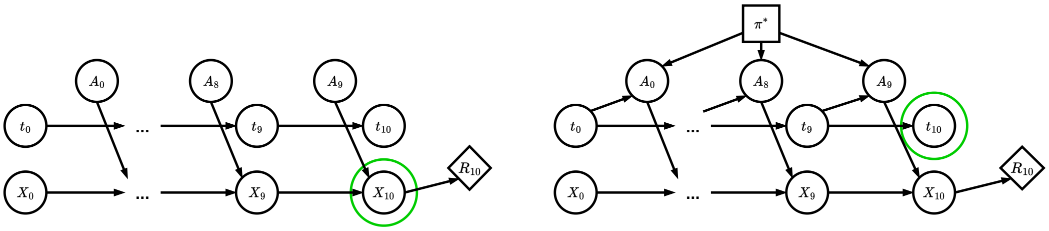

Figure 8: Two DBNs representing the dynamics of the watch-the-time environment, when actions are sampled at random (left), and when they are determined by the optimal policy (right).

C Further Related Work

Early evidence of policy confounding Although to the best of our knowledge, we are the first to bring forward and describe mathematically the idea of policy confounding, a few prior works have reported evidence of particular forms of policy confounding. In their review of the Arcade Learning Environment (ALE; Bellemare et al., 2013), Machado et al. (2018) explain that because the games are fully deterministic (i.e., initial states are fixed and transitions are deterministic), open-loop policies that memorize good action sequences can achieve high scores in ALE. Clearly, this can only occur if the policies themselves are also deterministic. In such cases, policies, acting as confounders, induce a spurious correlation between the past action sequences and the environment states. Similarly, Song et al. (2020) showed, by means of saliency maps, how agents may learn to use irrelevant features of the environment that happen to be correlated with the agent's progress, such as background clouds or the game timer, as clues for outputting optimal actions. In this case, the policy is again a confounder for all these, a priori irrelevant, features. Zhang et al. (2018b) provide empirical results showing how large neural networks may overfit their training environments and, even when trained on a collection of procedurally generated environments, memorize the optimal action for each observation. Zhang et al. (2018a) shows how, when trained on a small subset of trajectories, agents fail to generalize to a set of test trajectories generated by the same simulator. Lan et al. (2023) report evidence of well-trained agents failing to perform well on Mujoco environments when starting from trajectories (states) that are out of the distribution induced by the agent's policy. We conceive this as a simple form of policy confounding. Since the Mujoco environments are also deterministic, agents following a fixed policy can memorize the best actions to take for each state instantiation, potentially relying on superfluous features. Hence, they can overfit to unnatural postures that would not occur under different policies. Finally, Nikishin et al. (2022) describe a phenomenon named 'primacy bias', which prevents agents trained on poor trajectories from further improving their policies. The authors show that this issue is particularly relevant when training relies heavily on early data coming from a fixed random policy. We hypothesize that one of the causes for this is also policy confounding. The random policy may induce spurious correlations that lead to the formation of rigid state (state) representations that are hard to recover from.

Generalization Generalization is a hot topic in machine learning. The promise of a model performing well in contexts other than those encountered during training is undoubtedly appealing. In the realm of reinforcement learning, the majority of research focuses on generalization to environments that, despite sharing a similar structure, differ somewhat from the training environment (Kirk et al., 2023). These differences range from small variations in the transition dynamics (e.g., sim-to-real transfer; Higgins et al., 2017; Tobin et al., 2017; Peng et al., 2018; Zhao et al., 2020), changes in the observations (i.e., modifying irrelevant information, such as noise: Mandlekar et al., 2017; Ornia et al., 2022, or background variables: Zhang et al., 2020; Stone et al., 2021), to alterations in the reward function, resulting in different goals or tasks (Taylor & Stone, 2009; Lazaric, 2012; Muller-Brockhausen et al., 2021). Instead, we focus on the problem of OOT generalization. We aim to ensure that agents perform effectively when confronted with situations that differ from those encountered along their usual trajectories. Note that, in our experiments agents are evaluated in altered environments with different dynamics than those seen during training. These alterations are only intended to force the agent to take different trajectories. Importantly, the trajectories we force the agent to take are possible in the original environment.

State abstraction State abstraction is concerned with removing from the representation all that state information that is irrelevant to the task. In contrast, we are worried about learning representations containing too little information, which can lead to state aliasing. Nonetheless, as argued by McCallum (1995), state abstraction and state aliasing are two sides of the same coin. That is why we borrowed the mathematical frameworks of state abstraction to describe the phenomenon of policy confounding. Li et al. (2006) provide a taxonomy of the types of state abstraction and how they relate to one another. Givan et al. (2003) introduce the concept of bisimulation, which is equivalent to our definition of Markov state representation (Definition 4). Ferns et al. (2006) proposes a method for measuring the similarity between two states. Castro (2020) notes that this metric is prohibitively expensive and suggests using a relaxed version that computes state similarity relative to a given policy. This is similar to our notion of π-Markov state representation (Definition 7). While the end goal of this metric is to group together states that are similar under a given policy, here we argue that this may lead to poor OOT generalization.

D Dynamic Bayesian Networks

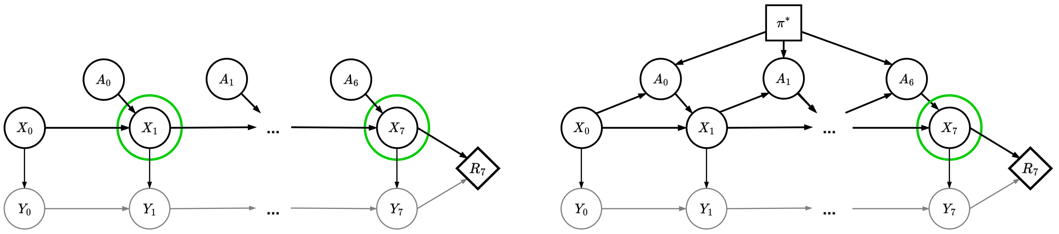

Figure 9: Two DBNs representing the dynamics of the Key2Door environment, when actions are sampled at random (left), and when they are determined by the optimal policy (right). The nodes labeled as X represent the agent's location, while the nodes labeled as Y represent whether or not the key has been collected. The agent can only see X. Hence, when actions that are sampled are random (left), the agent must remember its past locations to determine the reward R_7. Note that only X_1 and X_7 are highlighted in the left DBN. However, other variables in $\langle X_2, ..., X_6 \rangle$ might be needed, depending on when the key is collected. In contrast, when following the optimal policy, only X_7 is needed. In this second case, knowing the current location is sufficient to determine whether the key has been collected.

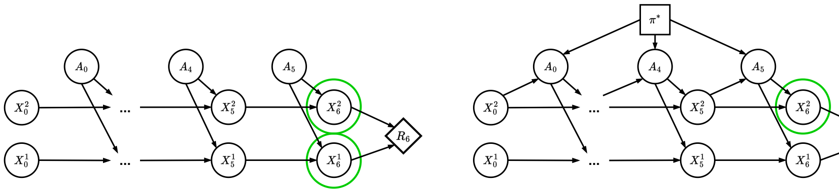

Figure 10: Two DBNs representing the dynamics of the Diversion environment, when actions are sampled at random (left), and when they are determined by the optimal policy (right). The nodes labeled as X^1 indicate the row where the agent is located; the nodes labeled as X^2 indicate the column. We see that when actions are sampled at random, both X_6^1 and X_6^2 are necessary to determine R_6. However, when actions are determined by the optimal policy, X_6^2 is sufficient, as the agent always stays at the top row.

E Experimental Results

E.1 Learned state representations

The results reported in Section 7 show that the OOT generalization problem exists. However, some may still wonder if the underlying reason is truly policy confounding. To confirm this, we compare the outputs of the policy at every state in the Frozen T-Maze when being fed the same states (observation stack) but two different signals. That is, we permute the variable containing the signal (X in the diagram of Figure 2) and leave the rest of the variables in the observation stack unchanged. We then feed the two versions to the policy network and measure the KL divergence between the two output probabilities. This metric is a proxy for how much the agent attends to the signal in every state. The heatmaps in Figure 11 show the KL divergences at various points during training (0, 10K, 30K, and 100K timesteps) when the true signal is 'green' and we replace it with 'purple'. We omit the two goal states since no actions are taken there. We see that initially (top left heatmap), the signal has very little influence on the policy (note the scale of the colormap is 10^{-6}), after 10K steps, the agent learns that the signal is very important when at the top right state (top right heatmap). After this, we start seeing how the influence of the signal at the top right state becomes less strong (bottom left heatmap) until it eventually disappears (bottom right heatmap). In contrast, the influence of the signal at the initial state becomes more and more important, indicating that after taking the first action, the agent ignores the signal and only attends to its own location. The results for the alternative case, 'purple' signal being replaced by 'green' signal, are shown in Figure 12.

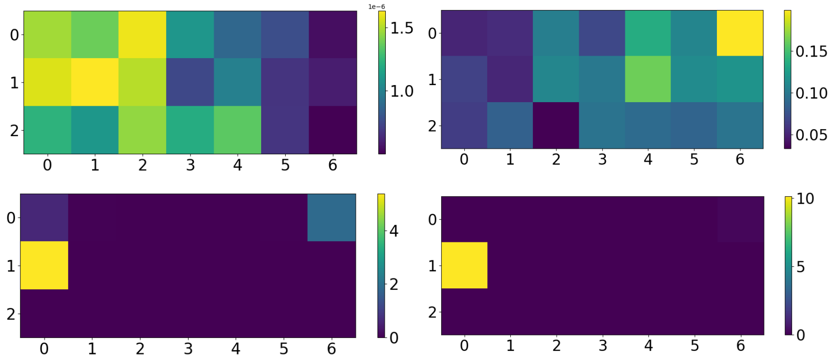

Figure 11: A visualization of the learned state representations. The heatmaps show the KL divergence between the action probabilities when feeding the policy network a stack of the past 10 observations and when feeding the same stack but with the value of the signal being switched from green to purple, after 0 (top left), 10K (top right), 30K (bottom left), and 100K (bottom right) timesteps of training.

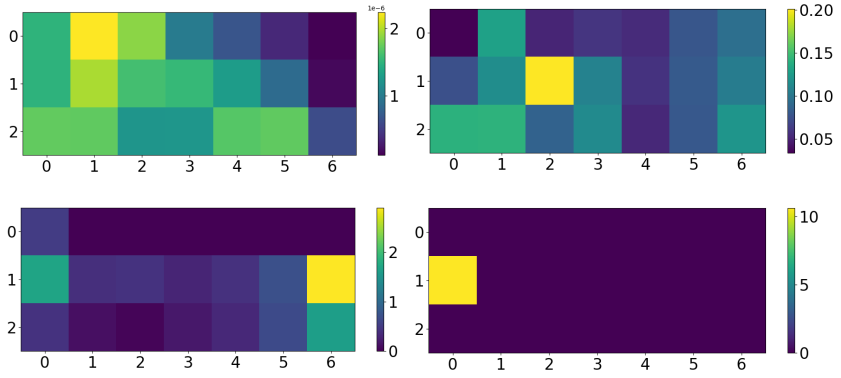

Figure 12: A visualization of the learned state representations. The heatmaps show the KL divergence between the action probabilities when feeding the policy network a stack of the past 10 observations and when feeding the same stack but with the value of the signal being switched from purple to green, after 0 (top left), 10K (top right), 30K (bottom left), and 100K (bottom right) timesteps of training.

E.2 Buffer size and exploration/domain randomization

Figures 13 and 14 report the results of the experiments described in Section 7 (paragraphs 2 and 3) for Key2Door and Diversion. We see how the buffer size also affects the performance of DQN in the two environments (left plots). We also see that exploration/domain randomization does improve OOT generalization in Diversion but not in Key2Door.

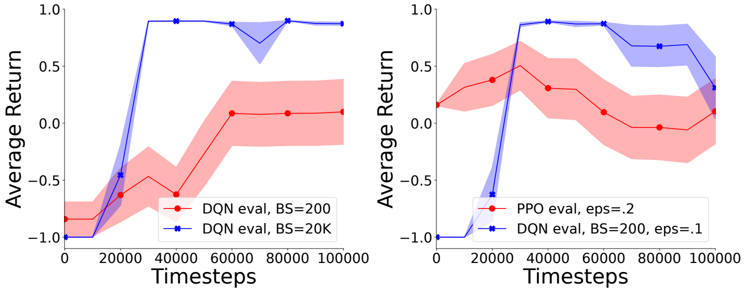

Figure 13: Key2Door. Left: DQN small vs. large buffer sizes. Right: PPO and DQN when adding stochasticity.

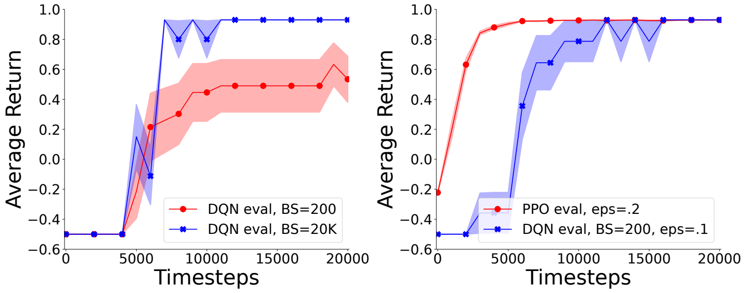

Figure 14: Diversion. Left: DQN small vs. large buffer sizes. Right: PPO and DQN when adding stochasticity.

F Further Experimental Details

We ran our experiments on an Intel i7-8650U CPU with 8 cores. Agents were trained with Stable Baselines3 (Raffin et al., 2021). Most hyperparameters were set to their default values except for the ones reported in Tables 1 (PPO) and 2 (DQN), which seemed to work better than the default values.

Table 1: PPO hyperparameters.

Rollout steps	128
Batch size	32
Learning rate	2.5e-4
Number epoch	3
Entropy coefficient	1.0e-2
Clip range	0.1
Value coefficient	1
Number Neurons 1st layer	128
Number Neurons 2nd layer	128

Table 2: DQN hyperparameters.

Buffer size	1.0e5
Learning starts	1.0e3
Learning rate	2.5e-4
Batch size	256
Initial exploration bonus	1.0
Final exploration bonus	0.0
Exploration fraction	0.2
Train frequency	5
Number Neurons 1st layer	128
Number Neurons 2nd layer	128