Training a Resilient Q-Network against Observational Interference

Chao-Han Huck Yang[1], **I-Te Danny Hung**[2], **Yi Ouyang**[3], **Pin-Yu Chen**[4]

Georgia Institute of Technology[1], Columbia University[2], Preferred Networks America[3], IBM Research AI[4]
huckiyang@gatech.edu, ih2320@columbia.edu, ouyangyi@gmail.com, pin-yu.chen@ibm.com

Abstract

Deep reinforcement learning (DRL) has demonstrated impressive performance in various gaming simulators and real-world applications. In practice, however, a DRL agent may receive faulty observation by abrupt interferences such as black-out, frozen-screen, and adversarial perturbation. How to design a resilient DRL algorithm against these rare but mission-critical and safety-crucial scenarios is an essential yet challenging task. In this paper, we consider a deep q-network (DQN) framework training with an auxiliary task of observational interferences such as artificial noises. Inspired by causal inference for **observational interference**, we propose a causal inference based DQN algorithm called causal inference Q-network (CIQ). We evaluate the performance of CIQ in several benchmark DQN environments with different types of interferences as auxiliary labels. Our experimental results show that the proposed CIQ method could achieve higher performance and more resilience against observational interferences.

Introduction

Deep reinforcement learning (DRL) methods have shown enhanced performance, gained widespread applications (Mnih et al. 2015, 2016; Silver et al. 2017), and improved robot learning (Gu et al. 2017) in navigation systems (Tai, Paolo, and Liu 2017; Nagabandi et al. 2018). However, most successful demonstrations of these DRL methods are usually trained and deployed under well-controlled situations. In contrast, real-world use cases often encounter inevitable observational uncertainty (Grigorescu et al. 2020; Hafner et al. 2018; Moreno et al. 2018) from an external attacker (Huang et al. 2017) or noisy sensor (Fortunato et al. 2018; Lee et al. 2018). For examples, playing online video games may experience sudden black-outs or frame-skippings due to network instabilities, and driving on the road may encounter temporary blindness when facing the sun. Such **an abrupt interference on the observation could cause serious issues** for DRL algorithms. Unlike other machine learning tasks that involve only a single mission at a time (e.g., image classification), an RL agent has to deal with a dynamic (Schmidhuber 1992) or even learn from latent states with generative models (Schmidhuber 1991; Jaderberg et al. 2017; Ha and Schmidhuber 2018; Hafner et al. 2018; Lynch et al. 2020) to anticipate future rewards in complex environments. Therefore, DRL-based systems are likely to propagate and even enlarge risks (e.g., delay and noisy pulsed-signals on sensor-fusion (Yurtsever et al. 2020; Johansen et al. 2015)) induced from the uncertain interference.

In this paper, we investigate the *resilience* ability of an RL agent to withstand unforeseen, rare, adversarial and potentially catastrophic interferences, and to recover and adapt by improving itself in reaction to these events. We consider a resilient generative RL framework with observational interferences as an auxiliary task. At each time, the agent's observation is subjected to a type of sudden interference at a predefined possibility. Whether or not an observation has interfered is referred to as the interference label.

Specifically, to train a resilient agent, we provide the agent with the interference labels during training. For instance, the labels could be derived from some uncertain noise generators recording whether the agent observes an intervened state at the moment as a binary causation label. By applying the labels as an *intervention* into the environment, the RL agent is asked to learn a binary causation label and embed a latent state into its model. However, when the trained agent is deployed in the field (i.e., the testing phase), the agent only receives the interfered observations but is agnostic to interference labels and needs to act resiliently against the interference.

For an RL agent to be resilient against interference, the agent needs to diagnose observations to make the correct inference about the reward information. To achieve this, the RL agent has to reason about what leads to desired rewards despite the irrelevant intermittent interference. To equip an RL agent with this reasoning capability, we exploit the causal inference framework. Intuitively, a causal inference model for observation interference uses an unobserved confounder (Pearl 2009, 2019, 1995b; Saunders et al. 2018; Bareinboim, Forney, and Pearl 2015; Zhang, Zhang, and Li 2020; Khemakhem et al. 2021) to capture the effect of the interference on the rewards (outcomes) collected from the environment. In recent works, RL is also showing additional benefits incorporating generative causal modeling, such as providing interpretability (Madumal et al. 2020), treatment estimation (Zhang and Bareinboim 2020, 2021), imitation learning (Zhang, Kumor, and Bareinboim 2020), enhanced invariant prediction (Zhang et al. 2020), and generative model

for transfer learning (Killian, Ghassemi, and Joshi 2020).

When such a confounder is available, the RL agent can focus on the confounder for relevant reward information and make the best decision. As illustrated in Figure 1, we propose a causal inference based DRL algorithm termed causal inference Q-network (CIQ). During training, when the interference labels are available, the CIQ agent will implicitly learn a causal inference model by embedding the confounder into a latent state. At the same time, the CIQ agent will also train a Q-network on the latent state for decision making. Then at testing, the CIQ agent will make use of the learned model to estimate the confounding latent state and the interference label. The design of CIQ is inspired by causal inference on state variable and using treatment switching method (Shalit, Johansson, and Sontag 2017) to learn latent variable by incorporating observational interference.

The history of latent states is combined into a causal inference state, which captures the relevant information for the Q-network to collect rewards in the environment despite of the observational interference.

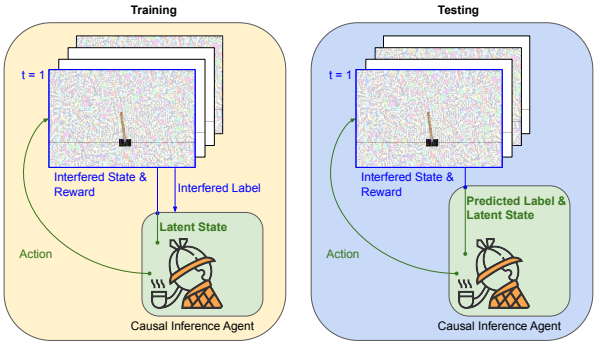


Figure 1: The proposed causal inference Q-network (CIQ) training and test framework, where the latent state is an unobserved (hidden) confounder variable. We refer the readers to Figure 3 for detailed descriptions on its graphical model.

In this paper, we evaluate the performance of our method in four environments: 1) Cartpole-v0 – the continuous control environment (Brockman et al. 2016); 2) the 3D graphical Banana Collector (Juliani et al. 2018)); 3) an Atari environment LunarLander-v2 (Brockman et al. 2016), and 4) pixel Cartpole – visual learning from the pixel inputs of Cartpole. For each of the environments, we consider four types of interference: (a) black-out, (b) Gaussian noise, (c) frozen screen, and (d) additive noise from adversarial perturbation.

In the testing phase mimicking the practical scenario that the agent may have interfered observations but is unaware of the true interference labels (i.e., happens or not), the results show that our CIQ method can perform better and more resilience against all the four types of interference. Furthermore, to benchmark the level of resilience of different RL models, we propose a new robustness measure, called CLEVER-Q, to evaluate the robustness of Q-network based RL algorithms. The idea is to compute a lower bound on the observation noise level such that the greedy action from the Q-network will remain the same against any noise below the lower bound. According to this robustness analysis, our CIQ algorithm indeed achieves higher CLEVER-Q scores compared with the baseline methods.

The main contributions of this paper include 1) a framework to evaluate the resilience of DQN-based DRL methods under abrupt observational interferences; 2) the proposed CIQ architecture and algorithm towards training a resilient DQN agent, and 3) an extreme-value theory based robustness metric (CLEVER-Q) for quantifying the resilience of Q-network based RL algorithms.

Related Works

Causal Inference for Generative Reinforcement Learning: Causal inference (Greenland, Pearl, and Robins 1999; Pearl 2009; Pearl, Glymour, and Jewell 2016; Pearl 2019; Robins, Rotnitzky, and Zhao 1995) has been used to empower the learning process under noisy observation and have better interpretability on deep learning models (Shalit, Johansson, and Sontag 2017; Louizos et al. 2017), also with efforts (Jaber, Zhang, and Bareinboim 2019; Forney, Pearl, and Bareinboim 2017; Bareinboim, Forney, and Pearl 2015; Bennett et al. 2021; Jung, Tian, and Bareinboim 2021) on causal online learning and bandit methods. Defining causation and applying causal inference framework to DRL still remains relatively unexplored. Recent works (Lu, Schölkopf, and Hernández-Lobato 2018; Tennenholtz, Mannor, and Shalit 2019) study this problem by defining action as one kind of intervention and estimating the causal effects. In contrast, we introduce observational interference into generative DRL by applying extra noisy and uncertain inventions. Inspired by the treatment switching and representation learning models (Shalit, Johansson, and Sontag 2017; Louizos et al. 2017; Helwegen, Louizos, and Forré 2020), we leverage the causal effect of observational interferences on states, and design an end-to-end structure for learning a *causal-observational* representation evaluating treatment effects on rewards.

Adversarial Perturbation: An intensifying challenge against deep neural network based systems is adversarial perturbation for making incorrect decisions. Many gradient-based noise-generating methods (Goodfellow, Shlens, and Szegedy 2015; Huang et al. 2017; Everett 2021) have been conducted for misclassification and mislead an agent's output action. As an example of using DRL model playing Atari games, an adversarial attacker (Lin et al. 2017; Yang et al. 2020c) could jam in a timely and barely detectable noise to maximize the prediction loss of a Q-network and cause massively degraded performance.

Partially Observable Markov Decision Processes (POMDPs): Our resilient RL framework can be viewed as a POMDP with interfered observations. Belief-state methods are available for simple POMDP problems (e.g., plan graph and the tiger problem (Kaelbling, Littman, and Cassandra 1998)), but no provably efficient algorithm is available for general POMDP settings (Papadimitriou and Tsitsiklis 1987; Gregor et al. 2018). Recently, Igl *et. al* (Igl et al. 2018) have proposed a DRL approach for POMDPs by combining variational autoencoder and policy-based learning, but this kind of methods do not consider the interference labels available during training in our resilient RL framework.

Resilient Reinforcement Learning

In this section, we formally introduce our resilient RL framework and provide an extreme-value theory based metric called CLEVER-Q for measuring the robustness of DQN-based methods.

We consider a sequential decision-making problem where an agent interacts with an environment. At each time t, the agent gets an observation x_t, e.g. a frame in a video environment. As in many RL domains (e.g., Atari games), we view $s_t = (x_{t-M+1}, \ldots, x_t)$ to be the state of the environment where M is a fixed number for the history of observations. Given a stochastic policy π, the agent chooses an action $a_t \sim \pi(s_t)$ from a discrete action space based on the observed state and receives a reward r_t from the environment. For a policy π, define the Q-function $Q^\pi(s,a) = \mathbb{E}\big[\sum_{t=0}^{\infty} \gamma^t r_t | s_0 = s, a_0 = a, \pi\big]$ where $\gamma \in (0,1)$ is the discount factor. The agent's goal is to find the optimal policy π^* that achieves the optimal Q-function given by $Q^*(s,a) = \max_\pi Q^\pi(s,a)$.

Resilience base on an Interventional Perspective

To evaluate the resilience ability of RL agents, we introduce additional interference as auxiliary information (as illustrated in Fig 1) as an empirical process (Pearl 2009; Louizos et al. 2017) for observation. Given a type of interference $\mathcal{I}$, the agent's observation becomes:

$$x'_t = F^{\mathcal{I}}(x_t, i_t) = i_t \times \mathcal{I}(x_t) + (1 - i_t) \times x_t \qquad (1)$$

where $i_t \in \{0, 1\}$ is the label indicating whether the observation is interfered at time t or not, and $\mathcal{I}(x_t)$ is the interfered observation.

We assume that interference labels i_t follow an i.i.d. Bernoulli process with a fixed interference probability $p^{\mathcal{I}}$ as a noise level.[1] For example, when $p^{\mathcal{I}}$ equals to 10%, each observational state has a 10% chance to be intervened under a perturbation. In this work, we consider the original observations, as illustrated in Figure 2 (a), under four types of interference as described below.

Gaussian Noise. Gaussian noise or white noise is a common interference to sensory data (Osband et al. 2019; Yurtsever et al. 2020). The interfered observation becomes $\mathcal{I}(x_t) = x_t + n_t$ with a zero-mean Gaussian noise n_t. The noise variance is set to be the variance of all recorded states as illustrated in Figure 2 (b).

Adversarial Observation. Following the standard adversarial RL attack setting, we use fast gradient sign method (FGSM) (Szegedy et al. 2014) to generate adversarial patterns against the DQN loss (Huang et al. 2017) as illustrated in Figure 2 (c). The observation is given by $\mathcal{I}(x_t) = x_t + \epsilon \operatorname{sign}(\nabla_{x_t} Q(x_t, y; \theta))$ where y is the optimal action by weighting over possible actions.

Observation Black-Out. Off-the-shelf hardware can affect the entire sensor networks as a sensing background (Yurtsever et al. 2020) over-shoot with $\mathcal{I}(x_t) = 0$ (Yan, Xu, and Liu 2016). This perturbation is realistic owing to overheat hardware and losing the observational information of sensors.

Frozen Frame. Lagging and frozen frame(s) (Kalashnikov et al. 2018) often come from limited data communication bottleneck bandwidth. A frozen frame is given by $\mathcal{I}(x_t) = x_{t-1}$. If the perturbation is constantly present, the frame will remain the first frozen frame since the perturbation happened.

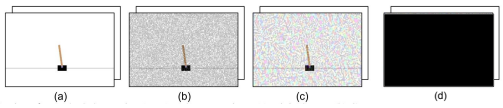


Figure 2: Visualization of perturbed observation (state) under uncertainty: (a) original state; (b) Gaussian perturbation; (c) adversarial perturbation (Huang et al. 2017), and (d) black-out perturbation (a white-out ablation in the Appendix E).

Resilient Reinforcement Learning Framework

With observational interference, instead of the actual state s_t, the agent only observes $s'_t = (x'_{t-M+1}, \ldots, x'_t)$. The agent now needs to choose its actions $a_t \sim \pi(s'_t)$ based on the interfered observation. The resilient RL objective for the agent is to find a policy π to maximize rewards in this environment under observational interference. Under the resilient framework, the goal of a Q-learning based agent is to learn the relation between s'_t and Q_t where $Q_t(a) = \max_\pi \mathbb{E}\big[\sum_{\tau=t}^{\infty} \gamma^{(\tau-t)} r_\tau | s'_t, a_t = a, \pi\big]$ denotes the Q-values given the interfered observation s'_t at time t.

From the RL model and the observation model of Eq. (1), the relation among the observation s'_t, Q-values Q_t, and interference i_t can be described by a causal graphical model (CGM) in Figure 3. In the CGM, $z_t = (s_t, i_{t-M+1}, \ldots, i_t)$ includes the actual state s_t of the system together with the interference labels which causally affects all s'_t, Q_t, and i_t. Note that z_t is not observable to the agent due to the interference; z_t could be viewed as a hidden confounder in causal inference.

Since only the interfered observation s'_t is available, the interference label i_t is also non-observable in evaluating the resilience ability of an agent. However, the interference information is often accessible in the training phase, such as the use of a navigation simulator recorded with noisy augmentation (Grigorescu et al. 2020) for simulating interference in the training environment. We will discuss in the next subsection the benefit of utilizing the interference labels to improve learning efficiency.

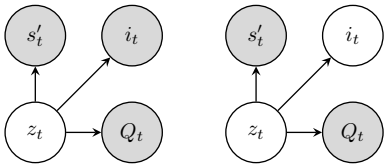
Learning with Interference Labels

The goal of a resilient RL agent is to learn $P(Q_t | s'_t)$ to infer the Q-value Q_t based on the interfered observation s'_t. Note that one can compute $P(Q_t | s'_t)$ by determining the joint distribution $P(z_t, s'_t, i_t, Q_t)$ of all variables in the CGM in Figure 3. Despite the presence of the hidden variable z_t, similar to causal inference with hidden confounders (Louizos et al. 2017), estimating the joint distribution $P(z_t, s'_t, i_t, Q_t)$ could be done efficiently when the agent is provided the interference labels i_t during training. On the other hand, if only the observation s'_t is available, the agent can only

[1] The i.i.d. assumption could be extended to a Markovian dynamic interference model. We show experiments with dynamic interference in Appendix E.

directly estimate $P(Q_t|s'_t)$, which is less efficient in terms of training sample usage.

We provide the interference type $\mathcal{I}$ and the interference labels i_t to efficiently train a resilient RL agent as shown in Figure 3(b); however, in the actual testing environment, the agent only has access to the interfered observations x'_t as in Figure 3(a).



(a) Training (b) Testing

Figure 3: Causal graphical model (CGM) for the training phase (a) and the testing phase (b). White nodes s'_t and Q_t are observable. Node $z_t = (s_t, i_{t-M+1}, \ldots, i_t)$, colored by white, is not observable. Node i_t, colored by white in (b), is only observable during training.

Causal Inference Q-Network

With the observable variables (s'_t, i_t, Q_t) in Figure 3(a) during training, we aim to learn a model to infer the Q-values by estimating the joint distribution $P(z_t, s'_t, i_t, Q_t)$. Despite the underlying dynamics in the RL system, when we view the interference as a treatment, the CGM in Figure 3(a) resembles some common causal inference models with binary treatment information and hidden confounders (Louizos et al. 2017). In this kind of causal inference problems, by leveraging on the binary property for treatment information, TARNet (Shalit, Johansson, and Sontag 2017) and CEVAE (Louizos et al. 2017) introduced a binary switching neural architecture to efficiently learn latent models for causal inference.

Inspired by the switching mechanism for causal inference, we propose the causal inference Q-network, referred as CIQ, that maps the interfered observation s'_t into a latent state z_t, makes proper inferences about the interference condition i_t, and adjusts its policy based on the estimated interference.

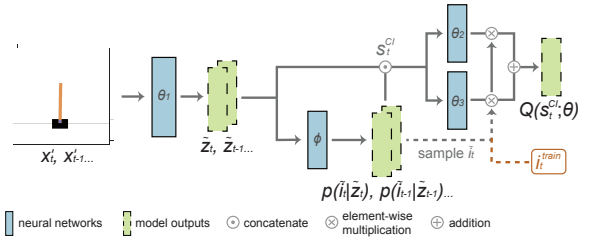


Figure 4: CIQ architecture. The notation i_t^{train} denotes the inference label available during training, whereas $\tilde{i}_t$ is sampled during inference as i_t is unknown.

We approximate the latent state by a neural network $\tilde{z}_t = f_1(x'_t; \theta_1)$. From the latent state, we generate the estimated interference label $\tilde{i}_t \sim p(\tilde{i}_t|z_t) = f_I(z_t; \phi)$. We denote $s_t^{CI} = (\tilde{z}_{t-M+1}, \tilde{i}_{t-M+1}, \ldots, \tilde{z}_t, \tilde{i}_t)$ to be the causal inference state. As discussed in the previous subsection, the causal inference state acts as a confounder between the interference and the reward. Therefore, instead of using the interfered state s'_t, the causal inference state s_t^{CI} contains more relevant information for the agent to maximize rewards. Using the causal inference state helps focus on meaningful and informative details even under interference.

With the causal inference state s_t^{CI}, the output of the Q-network $Q(s_t^{CI}; \theta)$ is set to be switched between two neural networks $f_2(s_t^{CI}; \theta_2)$ and $f_3(s_t^{CI}; \theta_3)$ by the interference label. Such a switching mechanism prevents our network from over-generalizing the causal inference state. During training, switching between the two neural networks is determined by the training interference label i_t^{train}. We assume that the true interference label is available in the training phase so $i_t^{\text{train}} = i_t$. In the testing, when i_t is not available, we use the predicted interference label $\tilde{i}_t$ as the switch to decide which of the two neural networks to use.

All the neural networks f_1, f_2, f_3, f_I have two fully connected layers[2] with each layer followed by the ReLU activation except for the last layer in f_2, f_3 and f_I. The overall CIQ model is shown in Figure 4 and $\theta = (\theta_1, \theta_2, \theta_3, \phi)$ denotes all its parameters. Note that, as common practice for discrete action spaces, the Q-network output $Q(s_t^{CI}; \theta)$ is an $\mathcal{A}$-dimensional vector where $\mathcal{A}$ is the size of the action space, and each dimension represents the value for taking the corresponding action.

Finally, we train the CIQ model $Q(s'_t; \theta)$ end-to-end by the DQN algorithm with an additional loss for predicting the interference label. The overall CIQ objective function is defined as:

$$
\begin{aligned}
&L^{\text{CIQ}}(\theta_1, \theta_2, \theta_3, \phi) = i_t^{\text{train}} \cdot L^{\text{DQN}}(\theta_1, \theta_2, \phi) \\
&+ (1 - i_t^{\text{train}}) \cdot L^{\text{DQN}}(\theta_1, \theta_3, \phi) + \lambda \cdot (i_t^{\text{train}} \log p(\tilde{i}_t|\tilde{z}_t; \theta_1, \phi) \\
&+ (1 - i_t^{\text{train}}) \log(1 - p(\tilde{i}_t|\tilde{z}_t; \theta_1, \phi))), \qquad (2)
\end{aligned}
$$

where λ is a scaling constant and is set to 1 for simplicity. Due to the design of the causal inference state and the switching mechanism, we will show that CIQ can perform resilient behaviors against the observation interferences. We introduce how to quantify the robustness of a Q-network under noisy observation in next subsection. The entire CIQ training procedure is described by Algorithm 1 in Appendix B.

CLEVER-Q: A Robustness Evaluation Metric for Q-Networks

Here we provide a comprehensive score (CLEVER-Q) for evaluating the robustness of a Q-network model by extending the CLEVER robustness score (Weng et al. 2018) designed for classification tasks to Q-network based DRL tasks. Consider an ℓ_p-norm bounded ($p \geq 1$) perturbation δ to the state s_t. We first derive a lower bound β_L on the minimal perturbation to s_t for altering the action with the top Q-value, i.e., the greedy action. For a given s_t and a Q-network, this lower

[2]Though such manner may lead to the myth of over-parameterization, our ablation study proves that we can achieve better results with almost the same amount of parameters.

bound β_L provides a robustness guarantee that the greedy action at s_t will be the same as that of *any* perturbed state $s_t + \delta$, as long as the perturbation level $\|\delta\|_p \leq \beta_L$. Therefore, the larger the value β_L is, the more resilience of the Q-network against perturbations can be guaranteed. Our CLEVER-Q score uses the extreme value theory to evaluate the lower bound β_L as a robustness metric for benchmarking different Q-network models. The proof of Theorem 1. is available in Appendix B.

Theorem 1. *Consider a Q-network $Q(s, a)$ and a state s_t. Let $\mathcal{A}^* = \arg\max_a Q(s_t, a)$ be the set of greedy (best) actions having the highest Q-value at s_t according to the Q-network. Define $g_a(s_t) = Q(s_t, \mathcal{A}^*) - Q(s_t, a)$ for every action a, where $Q(s_t, \mathcal{A}^*)$ denotes the best Q-value at s_t. Assume $g_a(s_t)$ is locally Lipschitz continuous[3] with its local Lipschitz constant denoted by L_q^a, where $1/p + 1/q = 1$ and $p \geq 1$. For any $p \geq 1$, define the lower bound*

$$\beta_L = min_{a \notin \mathcal{A}^*} g_a(s_t)/L_q^a. \tag{3}$$

Then for any δ such that $\|\delta\|_p \leq \beta_L$, we have $\arg\max_a Q(s_t, a) = \arg\max_a Q(s_t + \delta, a)$.

(a) Banana Collector (b) LunarLander-v2

Figure 5: Illustration of our environments on: (a) a 3D navigation task, banana collector (Juliani et al. 2018), and (b) a video game, LunarLander (Brockman et al. 2016).

Experiments

Environments for DQNs

Our testing platforms were based on (a) OpenAI Gym (Brockman et al. 2016), (b) Unity-3D environments (Juliani et al. 2018), (c) a 2D gaming environment (Brockman et al. 2016), and (d) visual learning from pixel inputs of cart pole. Our test environments cover some major application scenarios and feature discrete actions for training DQN agents with the CLEVER-Q analysis. For instance, Atari games and space-invaders are popular real-world applications. Unity 3D banana navigation is a physical simulator but provides virtual to real options for further implementations.

Vector Cartpole: Cartpole (Sutton et al. 1998) is a classical continuous control problem. We use Cartpole-v0 from Gym (Brockman et al. 2016) with a targeted reward $= 195.0$. The defined environment is manipulated by adding a force of $+1$ or -1 to a moving cart.

Banana Collector: The Banana collector shown in Figure 5 (a) is one of the Unity 3D baseline (Juliani et al. 2018). Different from the MuJoCo simulators with continuous actions, the Banana collector is controlled by four discrete actions corresponding to moving directions. The targeted reward is 12.0 points by accessing correct bananas $(+1)$. The state-space has 37 dimensions included velocity and a ray-based perception of objects around the agent.

Lunar Lander: Similar to the Atari gaming environments, Lunar Lander-v2 (Figure 5 (c)) is a discrete action environment from OpenAI Gym (Brockman et al. 2016) to control firing ejector with a targeted reward of 200. The state is an eight-dimensional vector that records the lander's position, velocity, angle, and angular velocities. The episode finishes if the lander crashes or comes to rest, receiving a reward -100 or $+100$ Firing ejector costs -0.3 each frame with $+10$ for each ground contact.

Pixel Cartpole: To further evaluate our models, we conduct experiments from the pixel inputs in the cartpole environment as a visual learning task. The size of input state is 400×600. We use a max-pooling and a convolution layer to extract states as network inputs. The environment includes two discrete actions $\{left, right\}$, which is identical to the Cartpole-v0 of the vector version.

Baseline Methods

In the experiments, we compare our CIQ algorithm with two sets of DQN-based DRL baselines to demonstrate the resilience capability of the proposed method. We ensure all the models have the **same number** of 9.7 millions **parameters** with careful fine-tuning to avoid model capacity issues.

Pure DQN: We use DQN as a baseline in our experiments. The DQN agent is trained and tested on interfered state s'_t. We also evaluate common DQN improvements in Appendix C and find the improvements (e.g., DDQN) have no significant effect against interference.

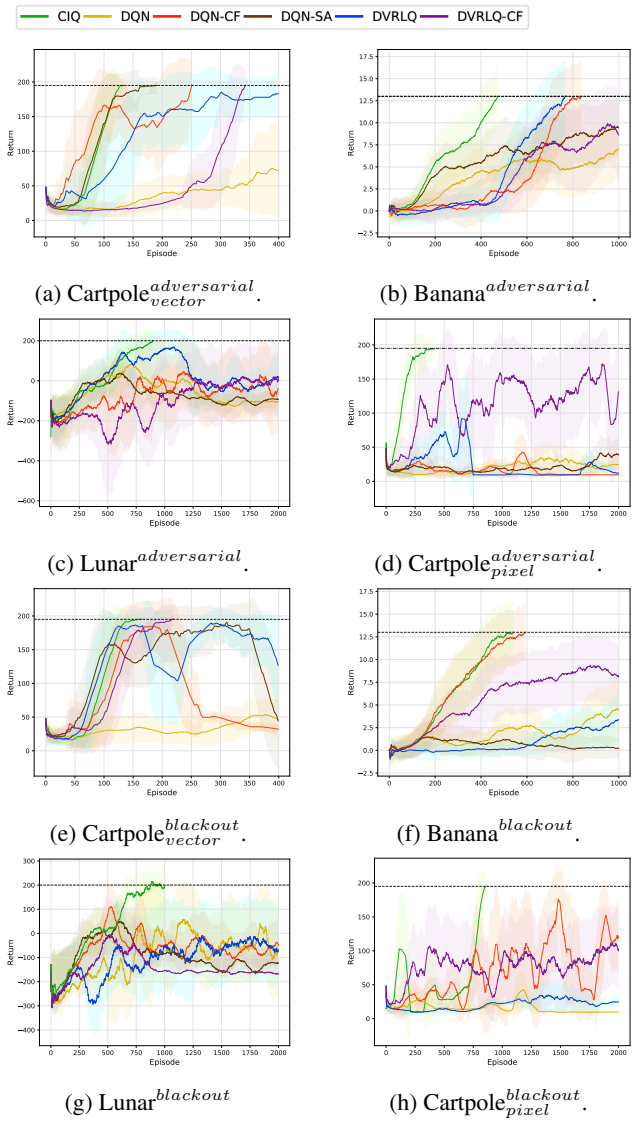
DQN with an interference classifier (DQN-CF): In the resilient reinforcement learning framework, the agent is given the true interference label i_t^{train} at training. Therefore, we would like to provide this additional information to the DQN agent for a **fair comparison.** During training, the interfered state s'_t is concatenated with the true label i_t^{train} as the input for the DQN agent. Since the true label is not available at testing, we train an additional binary classifier (CF) for the DQN agent. The classifier is trained to predict the interference label, and this predicted label will be concatenated with the interfered state as the input for the DQN agent during testing.

DQN with safe actions (DQN-SA): Inspired by shielding-based safe RL (Alshiekh et al. 2018), we consider a DQN baseline with safe actions (SA). The DQN-SA agent will apply the DQN action if there is no interference. However, if the current observation is interfered, it will choose the action used for the last uninterfered observation as the safe action. This action-holding method is also a typical control approach when there are missing observations (Franklin et al. 1998). Similar to DQN-CF, a binary classifier for interference is trained to provide predicted labels at testing.

DVRLQ and DVRLQ-CF: Motivated by deep variational RL (DVRL) (Igl et al. 2018), we provide a version of DVRL as a POMDP baseline. We call this baseline DVRLQ because we replace the A2C-loss with the DQN loss. Similar to DQN-CF, we also consider another baseline of DVRLQ with a

[3]Here locally Lipschitz continuous means $g_a(s_t)$ is Lipschitz continuous within the ℓ_p ball centered at s_t with radius R_p. We follow the same definition as in (Weng et al. 2018).

classifier, referred to as DVRLQ-CF, for a fair comparison using the interference labels.



(a) Cartpole$^{adversarial}_{vector}$. (b) Banana^{adversarial}.

(c) Lunar^{adversarial}. (d) Cartpole$^{adversarial}_{pixel}$.

(e) Cartpole$^{blackout}_{vector}$. (f) Banana^{blackout}.

(g) Lunar^{blackout} (h) Cartpole$^{blackout}_{pixel}$.

Figure 6: Performance of DQNs under potential (20%) adversarial and black-out interference.

Resilient RL on Average Returns

We run performance evaluation with six different interference probabilities (p^I in Sec.), including $\{0\%, 10\%, 20\%, 30\%, 40\%, 50\%\}$. We train each agent 50 times and highlight its standard deviation with lighter colors. Each agent is trained until the target score (shown as the dashed black line) is reached or until 400 episodes. We show the average returns for $p^I = 20\%$ under adversarial perturbation and black-out in Figure 6 and report the rest of the results in Appendix C.

CIQ (green) clearly outperforms all the baselines under all types of interference, validating the effectiveness of our CIQ in learning to infer and gaining resilience against a wide range of observational interferences. Pure DQN (yellow) cannot handle the interference with 20% noise level. DQN-CF (orange) and DQN-SA (brown) have competitive performance in some environments against certain interferences, but perform poorly in others. DVRLQ (blue) and DVRLQ-CF (purple) cannot achieve the target reward in most experiments and this might suggest the inefficiency of applying a general POMDP approach in a framework with a specific structure of observational interference.

Robustness Metrics based on Recording States

We evaluate the robustness of DQN and CIQ by the proposed CLEVER-Q metric. To make the test state environment consistent among different types and levels of interference, we record the interfered states, $S_N = \mathcal{I}(S_C)$, together with their clean states, S_C. We then calculate the average CLEVER-Q for DQN and CIQ based on the clean states S_C using Eq. 3 over 50 times experiments for each agent.

We also consider a retrospective robustness metric, the action correction rate (AC-Rate). Motivated by previous off-policy and error correction studies (Dulac-Arnold et al. 2012; Harutyunyan et al. 2016; Lin et al. 2017), AC-Rate is defined as the action matching rate $R_{Act} = \frac{1}{T}\sum_{t=0}^{T-1} \mathbf{1}_{\{a_t = a_t^*\}}$ between a_t and a_t^* over an episode with length T. Here a_t denotes the action taken by the agent with interfered observations S_N, and a_t^* is the action of the agent if clean states S_C were observed instead.

The roles of CLEVER-Q and AC-Rate are complementary as robustness metrics. CLEVER-Q measures sensitivity in terms of the margin (minimum perturbation) required for a given state to change the original action. AC-rate measures the utility in terms of action consistency. Altogether, they provide a comprehensive resilience assessment.

Table 1 reports the two robustness metrics for DQN and CIQ under two types of interference. CIQ attains higher scores than DQN in both CLEVER-Q and AC-Rate, reflecting better resilience in CIQ evaluations. We provide more robustness measurements in Appendix B and E.

Average Treatment Effect under Intervention

In a causal learning setting, evaluating treatment effects and conducting statistical refuting experiments are essential to support the underlying causal graphical model. Through resilient reinforcement learning framework, we could interpret DQN by estimating the average treatment effect (ATE) of each noisy and adversarial observation. We first define how to calculate a treatment effect in the resilient RL settings and conduct statistical refuting tests including random common cause variable test (T_c), replacing treatment with a random (placebo) variable (T_p), and removing a random subset of data (T_s). The open-source causal inference package Dowhy (Sharma, Kiciman et al. 2019) is used for analysis.

We refine a Q-network with discrete actions for estimating treatment effects based on Theorem 1 in (Louizos et al. 2017). In particular, individual treatment effect (ITE) can be defined as the difference between the two potential outcomes of a Q-network; and the average treatment effect (ATE) is the

Table 1: AC-Rate and CLEVER-Q robustness analysis under additive Gaussian (l_2-norm) and adversarial (l_∞-norm) perturbations on state in the vector Cartpole environment.

$\mathcal{I}$=L_2	AC-Rate		CLEVER-Q		$\mathcal{I}$=L_∞	AC-Rate		CLEVER-Q	
P%, $\mathcal{I}$	DQN	CIQ	DQN	CIQ	P%, $\mathcal{I}$	DQN	CIQ	DQN	CIQ
10%	82.10%	**99.61%**	0.176	**0.221**	10%	62.23%	**99.52%**	0.169	**0.248**
20%	72.15%	**98.52%**	0.130	**0.235**	20%	9.68%	**98.52%**	0.171	**0.236**
30%	69.74%	**98.12%**	0.109	**0.232**	30%	1.22%	**98.10%**	0.052	**0.230**

expected value of the potential outcomes over the subjects. In a binary treatment setting, for a Q-value function $Q_t(s_t)$ and the interfered state $\mathcal{I}(s_t)$, the ITE and ATE are calculated by:

$$Q_t^{ITE} = Q_t(s_t)\,(1-p_t) + Q_t(\mathcal{I}(s_t))p_t \quad (4)$$

$$ATE = \sum_{t=1}^{\mathcal{T}} \frac{\mathbb{E}\left[Q_t^{ITE}(\mathcal{I}(s_t))\right] - \mathbb{E}\left[Q_t^{ITE}(s_t)\right]}{\mathcal{T}} \quad (5)$$

where p_t is the estimated inference label by the agent and $\mathcal{T}$ is the total time steps of each episode. As expected, we find that CIQ indeed attains a better ATE and its significance can be informed by the refuting tests based on T_c, T_p and T_s. We refer to Appendix D for more details.

Additional analysis

We also conduct the following analysis to better understand our CIQ model. Environments with a dynamic noise level are evaluated. Due to the space limit, see their details in appendix C to E. Furthermore, a discussion on the advantage of sample complexity benefited from sequential learning with interference labels is included in Appendix C.

Neural saliency map: We apply the perturbation-based saliency map for DRL (Greydanus et al. 2018) as shown in Figure 7 and appendix to visualize the saliency centers of CIQ and others, which is based on the Q-value of each model as interpretable studies.

Treatment effect analysis: We provide treatment effect analysis on each kind of interference to statistically verify the CGM with lowest errors on average treatment effect refutation in appendix D.

Ablation studies: We conduct ablation studies by comparing several CIQ variants, each without a certain CIQ component, and verify the importance of the proposed CIQ architecture in Appendix E for future studies.

Test on different noise levels: We train CIQ under one noise level and test on another level, which shows that the difference in noise level does not affect much on the performance of CIQ model reported in Appendix C.

Transferability in robustness: Based on CIQ, we study how well can the robustness of different interference types transfer between training and testing environments. We evaluate two general settings (i) an identical interference type but different noise levels (Appendix E) and (ii) different interference types (Appendix E). Tab 3 summarizes the results.

Multiple interference types: We also provide a generalized version of CIQ that deals with multiple interference types in training and testing environments. Tab 2 summarizes the results. The generalized CIQ is equipped with a common encoder and individual interference decoders to study multi-module conditional inference, with some additional discussion in Appendix E.

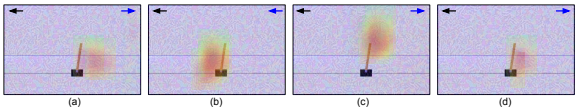


Figure 7: Perturbation-based saliency map on Pixel Cartpole under adversarial perturbation: (a) DQN, (b) CIQ, (c) DQN-CF; (d) DVRLQ-CF. The black arrows are correct actions and blue arrows are agents' actions. The neural saliency of CIQ makes more correct actions responding to ground actions.

Table 2: Stability test of proposed CIQ (*Train* Noise-Level, *Test* Noise-Level). We consider settings with different training and testing noise levels for CIQ evaluation. The (train, test)% case trains with train% noise then tests with test% noise. The results are similar to the cases with the same training and testing noise level in Table 5 in Appendix C.

Metrics	(10, 30)%	(30, 10)%	(30, 20)%	(30, 30)%	(30, 40)%	(30, 50)%
Performance	182.8	**195.0**	**195.0**	**195.0**	**195.0**	185.7
CLEVER-Q	0.195	0.239	0.232	0.230	0.224	0.215
AC-Rate	91.45 %	98.54%	98.62%	99.45%	98.45%	92.45%

Table 3: CIQ-MI: CIQ agent with an extended multi-interference (MI) architecture testing in Env_1 (noise level $P = 20\%$). As a proof of concept, we consider two interference types together, Gaussian noise and adversarial perturbation. In this setting every observation (state) can possibly undergo an interference with either Gaussian noise or Adversarial perturbation. CIQ-MI is capable of making correct action to solve (over 195.0) the testing environment when training with mixed interference types.

Train / **Test**	**Gaussian**	**Adversarial**	**Gaussian + Adversarial**
Gaussian	**195.1**	154.2	96.3
Adversarial	153.9	**195.0**	105.1
Gaussian + Adversarial	**195.0**	**195.0**	**195.0**

Conclusion

Our experiments suggest that, although some DQN-based DRL algorithms can achieve high scores under the normal condition, their performance can be severely degraded in the presence of interference. In order to be resilient against interference, we propose CIQ, a novel causal-inference-driven DRL algorithm. Evaluated on a wide range of environments and multiple types of interferences, the CIQ results show consistently superior performance over several RL baseline methods. We investigate the improved resilience of CIQ by CLEVER-Q and AC-Rate metrics. The CIQ code is available at `github.com/huckiyang/Obs-Causal-Q-Network`.

References

Alshiekh, M.; Bloem, R.; Ehlers, R.; Könighofer, B.; Niekum, S.; and Topcu, U. 2018. Safe reinforcement learning via shielding. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Bareinboim, E.; Forney, A.; and Pearl, J. 2015. Bandits with unobserved confounders: A causal approach. In *Advances in Neural Information Processing Systems*, 1342–1350.

Bennett, A.; Kallus, N.; Li, L.; and Mousavi, A. 2021. Off-policy evaluation in infinite-horizon reinforcement learning with latent confounders. In *International Conference on Artificial Intelligence and Statistics*, 1999–2007. PMLR.

Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; and Zaremba, W. 2016. Openai gym. *arXiv preprint arXiv:1606.01540*.

Dulac-Arnold, G.; Denoyer, L.; Preux, P.; and Gallinari, P. 2012. Fast reinforcement learning with large action sets using error-correcting output codes for mdp factorization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 180–194. Springer.

Everett, M. 2021. Neural Network Verification in Control. *arXiv preprint arXiv:2110.01388*.

Forney, A.; Pearl, J.; and Bareinboim, E. 2017. Counterfactual data-fusion for online reinforcement learners. In *International Conference on Machine Learning*, 1156–1164.

Fortunato, M.; Azar, M. G.; Piot, B.; Menick, J.; Osband, I.; Graves, A.; Mnih, V.; Munos, R.; Hassabis, D.; Pietquin, O.; et al. 2018. Noisy networks for exploration. *ICLR 2018, arXiv preprint arXiv:1706.10295*.

Franklin, G. F.; Powell, J. D.; Workman, M. L.; et al. 1998. *Digital control of dynamic systems*, volume 3. Addison-wesley Menlo Park, CA.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2015. Explaining and harnessing adversarial examples. *ICLR*.

Greenland, S.; Pearl, J.; and Robins, J. M. 1999. Causal diagrams for epidemiologic research. *Epidemiology*, 37–48.

Gregor, K.; Papamakarios, G.; Besse, F.; Buesing, L.; and Weber, T. 2018. Temporal difference variational auto-encoder. *arXiv preprint arXiv:1806.03107*.

Greydanus, S.; Koul, A.; Dodge, J.; and Fern, A. 2018. Visualizing and Understanding Atari Agents. In *International Conference on Machine Learning*, 1792–1801.

Grigorescu, S.; Trasnea, B.; Cocias, T.; and Macesanu, G. 2020. A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 37(3): 362–386.

Gu, S.; Holly, E.; Lillicrap, T.; and Levine, S. 2017. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *2017 IEEE international conference on robotics and automation (ICRA)*, 3389–3396. IEEE.

Ha, D.; and Schmidhuber, J. 2018. World models. *arXiv preprint arXiv:1803.10122*.

Hafner, D.; Lillicrap, T.; Fischer, I.; Villegas, R.; Ha, D.; Lee, H.; and Davidson, J. 2018. Learning latent dynamics for planning from pixels. *arXiv preprint arXiv:1811.04551*.

Harutyunyan, A.; Bellemare, M. G.; Stepleton, T.; and Munos, R. 2016. Q lamda with Off-Policy Corrections. In *International Conference on Algorithmic Learning Theory*, 305–320. Springer.

Helwegen, R.; Louizos, C.; and Forré, P. 2020. Improving Fair Predictions Using Variational Inference In Causal Models. *arXiv preprint arXiv:2008.10880*.

Huang, S.; Papernot, N.; Goodfellow, I.; Duan, Y.; and Abbeel, P. 2017. Adversarial attacks on neural network policies. *arXiv preprint arXiv:1702.02284*.

Igl, M.; Zintgraf, L.; Le, T. A.; Wood, F.; and Whiteson, S. 2018. Deep Variational Reinforcement Learning for POMDPs. In *International Conference on Machine Learning*, 2117–2126.

Jaber, A.; Zhang, J.; and Bareinboim, E. 2019. Causal identification under markov equivalence: Completeness results. In *International Conference on Machine Learning*, 2981–2989.

Jaderberg, M.; Mnih, V.; Czarnecki, W. M.; Schaul, T.; Leibo, J. Z.; Silver, D.; and Kavukcuoglu, K. 2017. Reinforcement learning with unsupervised auxiliary tasks. *ICLR*.

Johansen, T. A.; Cristofaro, A.; Sørensen, K.; Hansen, J. M.; and Fossen, T. I. 2015. On estimation of wind velocity, angle-of-attack and sideslip angle of small UAVs using standard sensors. In *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, 510–519. IEEE.

Juliani, A.; Berges, V.-P.; Vckay, E.; Gao, Y.; Henry, H.; Mattar, M.; and Lange, D. 2018. Unity: A general platform for intelligent agents. *arXiv preprint arXiv:1809.02627*.

Jung, Y.; Tian, J.; and Bareinboim, E. 2021. Estimating identifiable causal effects through double machine learning. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*.

Kaelbling, L. P.; Littman, M. L.; and Cassandra, A. R. 1998. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2): 99–134.

Kalashnikov, D.; Irpan, A.; Pastor, P.; Ibarz, J.; Herzog, A.; Jang, E.; Quillen, D.; Holly, E.; Kalakrishnan, M.; Vanhoucke, V.; et al. 2018. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. *arXiv preprint arXiv:1806.10293*.

Khemakhem, I.; Monti, R.; Leech, R.; and Hyvarinen, A. 2021. Causal autoregressive flows. In *International Conference on Artificial Intelligence and Statistics*, 3520–3528. PMLR.

Killian, T. W.; Ghassemi, M.; and Joshi, S. 2020. Counterfactually Guided Policy Transfer in Clinical Settings. *arXiv preprint arXiv:2006.11654*.

Lee, G.; Hou, B.; Mandalika, A.; Lee, J.; Choudhury, S.; and Srinivasa, S. S. 2018. Bayesian policy optimization for model uncertainty. *arXiv preprint arXiv:1810.01014*.

Lin, Y.-C.; Hong, Z.-W.; Liao, Y.-H.; Shih, M.-L.; Liu, M.-Y.; and Sun, M. 2017. Tactics of adversarial attack on deep reinforcement learning agents. *arXiv preprint arXiv:1703.06748*.

Louizos, C.; Shalit, U.; Mooij, J. M.; Sontag, D.; Zemel, R.; and Welling, M. 2017. Causal effect inference with deep latent-variable models. In *Advances in Neural Information Processing Systems*, 6446–6456.

Lu, C.; Schölkopf, B.; and Hernández-Lobato, J. M. 2018. Deconfounding reinforcement learning in observational settings. *arXiv preprint arXiv:1812.10576*.

Lynch, C.; Khansari, M.; Xiao, T.; Kumar, V.; Tompson, J.; Levine, S.; and Sermanet, P. 2020. Learning latent plans from play. In *Conference on Robot Learning*, 1113–1132. PMLR.

Madumal, P.; Miller, T.; Sonenberg, L.; and Vetere, F. 2020. Explainable reinforcement learning through a causal lens. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 2493–2500.

Mnih, V.; Badia, A. P.; Mirza, M.; Graves, A.; Lillicrap, T.; Harley, T.; Silver, D.; and Kavukcuoglu, K. 2016. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, 1928–1937.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; et al. 2015. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529.

Moreno, P.; Humplik, J.; Papamakarios, G.; Pires, B. A.; Buesing, L.; Heess, N.; and Weber, T. 2018. Neural belief states for partially observed domains. In *NeurIPS 2018 workshop on Reinforcement Learning under Partial Observability*.

Nagabandi, A.; Clavera, I.; Liu, S.; Fearing, R. S.; Abbeel, P.; Levine, S.; and Finn, C. 2018. Learning to adapt in dynamic, real-world environments through meta-reinforcement learning. *arXiv preprint arXiv:1803.11347*.

Osband, I.; Doron, Y.; Hessel, M.; Aslanides, J.; Sezener, E.; Saraiva, A.; McKinney, K.; Lattimore, T.; Szepezvari, C.; Singh, S.; et al. 2019. Behaviour suite for reinforcement learning. *arXiv preprint arXiv:1908.03568*.

Papadimitriou, C. H.; and Tsitsiklis, J. N. 1987. The complexity of Markov decision processes. *Mathematics of operations research*, 12(3): 441–450.

Pearl, J. 1995. On the testability of causal models with latent and instrumental variables. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, 435–443. Morgan Kaufmann Publishers Inc.

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

Pearl, J. 2019. The seven tools of causal inference, with reflections on machine learning. *Communications of the ACM*, 62(3): 54–60.

Pearl, J.; Glymour, M.; and Jewell, N. P. 2016. *Causal inference in statistics: A primer*. John Wiley & Sons.

Robins, J. M.; Rotnitzky, A.; and Zhao, L. P. 1995. Analysis of semiparametric regression models for repeated outcomes in the presence of missing data. *Journal of the american statistical association*, 90(429): 106–121.

Saunders, W.; Sastry, G.; Stuhlmueller, A.; and Evans, O. 2018. Trial without error: Towards safe reinforcement learning via human intervention. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, 2067–2069. International Foundation for Autonomous Agents and Multiagent Systems.

Schmidhuber, J. 1991. Reinforcement learning in Markovian and non-Markovian environments. In *Advances in neural information processing systems*, 500–506.

Schmidhuber, J. 1992. Learning complex, extended sequences using the principle of history compression. *Neural Computation*, 4(2): 234–242.

Shalit, U.; Johansson, F. D.; and Sontag, D. 2017. Estimating individual treatment effect: generalization bounds and algorithms. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 3076–3085. JMLR. org.

Sharma, A.; Kiciman, E.; et al. 2019. DoWhy A Python package for causal inference. *KDD 2019 workshop*.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. 2017. Mastering the game of Go without human knowledge. *Nature*, 550(7676): 354.

Sutton, R. S.; Barto, A. G.; Bach, F.; et al. 1998. *Reinforcement learning: An introduction*. MIT press.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2014. Intriguing properties of neural networks. *International Conference on Learning Representations*.

Tai, L.; Paolo, G.; and Liu, M. 2017. Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 31–36. IEEE.

Tennenholtz, G.; Mannor, S.; and Shalit, U. 2019. Off-Policy Evaluation in Partially Observable Environments. *arXiv preprint arXiv:1909.03739*.

Weng, T.-W.; Zhang, H.; Chen, P.-Y.; Yi, J.; Su, D.; Gao, Y.; Hsieh, C.-J.; and Daniel, L. 2018. Evaluating the robustness of neural networks: An extreme value theory approach. *arXiv preprint arXiv:1801.10578*.

Yan, C.; Xu, W.; and Liu, J. 2016. Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle. *DEFCON24*.

Yang, C.-H. H.; Qi, J.; Chen, P.-Y.; Ouyang, Y.; Hung, I.-T. D.; Lee, C.-H.; and Ma, X. 2020. Enhanced Adversarial Strategically-Timed Attacks Against Deep Reinforcement Learning. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3407–3411. IEEE.

Yurtsever, E.; Lambert, J.; Carballo, A.; and Takeda, K. 2020. A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access*, 8: 58443–58469.

Zhang, A.; Lyle, C.; Sodhani, S.; Filos, A.; Kwiatkowska, M.; Pineau, J.; Gal, Y.; and Precup, D. 2020. Invariant causal prediction for block mdps. In *International Conference on Machine Learning*, 11214–11224. PMLR.

Zhang, C.; Zhang, K.; and Li, Y. 2020. A Causal View on Robustness of Neural Networks. *Advances in Neural Information Processing Systems*, 33.

Zhang, J.; and Bareinboim, E. 2020. Designing optimal dynamic treatment regimes: A causal reinforcement learning approach. In *International Conference on Machine Learning*, 11012–11022. PMLR.

Zhang, J.; and Bareinboim, E. 2021. Bounding Causal Effects on Continuous Outcome. In *Proceedings of the 35nd AAAI Conference on Artificial Intelligence*.

Zhang, J.; Kumor, D.; and Bareinboim, E. 2020. Causal imitation learning with unobserved confounders. *Advances in Neural Information Processing Systems*, 33.