TOWARD OPTIMAL ADVERSARIAL ATTACKS IN REINFORCEMENT LEARNING VIA RATE–DISTORTION THEORY

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning (RL) has been deployed in many security-related applications. To improve the robustness of RL systems, it is important to systematically study potential adversarial attacks. Most prior work has considered deterministic adversarial attack strategies targeting either the agent or the environment, which can often be mitigated or reversed by defensive RL agents. In this paper, we propose a provably "uncounterable" class of adversarial attacks on RL systems. Our attack leverages a rate-distortion information-theoretic approach to *stochastically* perturb the agents' observations of the underlying Markov Decision Process (MDP). These perturbations are designed to strongly or even entirely limit the agent's ability to obtain accurate information about the true environment. This proposed framework is modular and can be integrated with various observation attack strategies targeting different components of the MDP, including state, actions, transition functions, or reward signals. We derive an information-theoretic lower bound on the victim agent's reward regret and show the impact of rate-distortion attacks on state-of-the-art algorithms.

1 INTRODUCTION

Reinforcement learning (RL) algorithms are used in many safety or security-related applications, such as autonomous driving Shalev-Shwartz et al. (2016), financial decisions Liu et al. (2020), recommendation systems Zhao et al. (2018), wireless communication Mastronarde & van der Schaar (2011), and also drone and robot algorithms Song et al. (2021). However, deploying RL in real-world settings raises important safety concerns. Studying potential adversarial attacks on RL systems can help evaluate the robustness of the performance of RL agents under perturbed conditions. This understanding can guide the design of defenses, reduce potential adversarial damage, and ultimately lead to more robust and secure RL systems.

Adversarial attacks on RL can be categorized based on the specific component targeted by the attacker. Common categories include observation attacks, action attacks, reward attacks, transition (environmental) attacks, as well as combinations of these Huang & Zhu (2019); Liu & Lai (2023); Zhang et al. (2020b); Liu & Lai (2021); Ma et al. (2019); Littman (1994); Xu et al. (2023); Huang & Zhu (2019); Behzadan & Munir (2017). Some attacks directly manipulate the agent's inputs or outputs, such as its observations or actions, while others change the environment's dynamics, including state transition laws or reward functions.

Most prior work focuses on deterministic attack strategies, where the victim agent observes a perturbed but deterministic environment. These approaches typically assume that the agent is unaware of the attack strategy. However, if the agent does suspect manipulations, it may learn a mapping between the corrupted signal and the true underlying signal, allowing it to recover the ground-truth property and to neutralize the attack. As a result, deterministic attacks are not guaranteed to remain effective against agents that are aware of the attack, and their success cannot be ensured under defensive strategies.

In contrast, we propose a provably "invincible" adversarial attack that replaces deterministic perturbations with probabilistic ones. Instead of showing the victim a fixed corrupted signal, the attack presents random observations drawn from a carefully designed distribution. This distribution is constructed so that the observed signal carries little to no information about the true underlying

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

property, whether it is the state, action, reward, or environment dynamics. An information-theoretic analysis guarantees this loss of information.

The key advantage of this information-theoretic approach is that, even if the victim agent fully understands the attack strategy, it still cannot infer the true property from the randomized observations. This inherent uncertainty prevents the agent from recovering the true environment and forces it to incur an unavoidable performance loss (regret). In this sense, the attack is "invincible": its effectiveness does not depend on the agent's awareness or defensive adaptations.

Motivating Example (Air Defense with Deceptive Signals): Consider a fully-observable environment involving two parties in an air defense scenario. An attacker (Party A) launches incoming objects toward a defender (Party B), who must decide how to allocate limited interception resources. At each step, the state consists of two parts: the remaining interception resources and the currently observed object. The object includes all available information from sensors and may be either a real missile or a decoy. The action is the amount of resources the defender chooses to use to intercept the current object.

In a traditional adversarial strategy, the attacker deterministically replaces real missiles with decoys to mislead the defender. However, such a strategy is ultimately predictable: Once the defender learns or infers this pattern, the observation becomes highly predictable about the attacker's behavior, for example, "All incoming objects are decoys" and the defender can adapt by ignoring them. In contrast, our rate-distortion type of attack operates by controlling the information content of the defender's observations. The attacker randomizes between real and fake objects under a budget constraint. For example, each observed object may correspond to a real missile or a decoy with comparable likelihood, making them statistically difficult to distinguish. As a result, the defender faces irreducible uncertainty about the true state of the environment, which often leads to unsuccessful defense or inefficient resource usage.

Contribution: In this paper, we propose a new information-theoretic rate-distortion approach that can be combined with any deterministic adversarial attacks on the RL systems. By applying this rate-distortion idea, attackers randomly, rather than deterministically, map the true quantities to perturbed ones. We provide a lower bound on the expected regret experienced by the victim RL agent, regardless of the defense strategies the victim adopts, through an information-theoretic analysis. To the best of our knowledge, this is among the first works to formulate adversarial attacks on RL algorithms using rate-distortion theory from the information theory perspective.

We want to emphasize that the evaluation of our randomized attacks is in the context of an *uncertain RL environment*, i.e., the environment dynamics is random with some prior probability distribution, which is different from most RL literature that assumes a fixed dynamics rule. Our numerical and theoretical results show that the rate-distortion information-theoretic adversarial attack can greatly reduce the victim agent's expected reward compared to deterministic attacks.

Related Works: In a nutshell, we propose a novel adversarial attack by injecting statistical uncertainties into the agent's observations, as inferred by the victim agent. In the context of adversarial attacks on reinforcement learning, adopted perturbations are primarily deterministic or optimization-based. For example, attackers in Zhang et al. (2020a) learn an optimized adversarial policy that outputs perturbations against the agent's policy. Earlier works like Pinto et al. (2017) develop attack strategies based on a two-layer zero-sum game between the agent and its adversary. Similarly, Yu et al. (2021) formulates robust RL as a constrained minimax game between the RL agent and an environmental adversary that models uncertainties such as model parameter variations and adversarial disturbances.

Even though the adversary's policies may be stochastic during exploration in the above works, the attack itself is optimized strategically to minimize reward. In contrast, the adversary considered in our paper statistically changes the victim agent's observation of the MDP into false observations following a ground-truth-observation-dependent distribution. None of the above works is based on the idea of mutual information minimization.

Franzmeyer et al. (2024) integrates information-theoretic detectability constraints to create stealthy adversarial attacks. However, they use information theory to constrain the statistical detectability of attacks during test-time, by bounding the KL-divergence between the attacked and original observation trajectories. In contrast, our rate-distortion attack focuses on attacking the observations randomly

during the training, aiming to minimize the mutual information between the agent's observed and ground-truth environment.

The rate-distortion driven type of adversarial attacks has been studied in the control literature. For example, Fawzi et al. (2014) studies sparse adversarial corruption in linear dynamical systems when fewer than half of the sensors are compromised. Kim & Lavaei (2025) investigates exact recovery guarantees for linear systems under adversarial state perturbations. These works primarily focus on continuous linear dynamical systems. In contrast, our work considers reinforcement learning and MDP, where the objective is not state recovery but policy learning under uncertainty.

2 Problem Formulation

We consider a discounted Markov decision process (MDP) $\mathcal{MG} = \{\mathcal{S}, \mathcal{A}, P, \gamma, R\}$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the state transition probability function, $\gamma \in [0, 1)$ is the discount factor, and $R : \mathcal{S} \times \mathcal{A} \to [0, 1]$ is the reward function. We study adversarial perturbations on an MDP component X, and $X \in \{\mathcal{S}, \mathcal{A}, P, R\}$.

Stochastic Observation Attacks An attacker does not directly modify the environment but instead perturbs the agent's perception of X. Specifically, the attacker induces a random variable Y (the perceived quantity) according to a conditional distribution

$$P(Y|X).$$

For simplicity, we assume $\Omega(Y) = \Omega(X)$, although the two spaces need not be the same.

Unlike prior work that considers deterministic perturbations $X \to Y$, we allow stochastic mappings. This prevents adaptive agents from reliably inferring or inverting the corruption over time, thereby sustaining the effectiveness of the attack.

2.1 Example: Transition Kernel Attack

We now focus on the case where the attacked quantity is the transition kernel, i.e., $X = P$.

We consider a random MDP $\{\mathcal{S}, \mathcal{A}, X, R, \gamma\}$, where the transition kernel X is sampled once from a prior $p(X)$ and then fixed throughout all episodes. Let $\Omega(X) = \{X_1, \ldots, X_N\}$ denote the finite set of possible transition kernels. For a fixed kernel X_i, define the value function of a policy π as

$$V^{\pi}_{X_i}(s) = \mathbb{E}_{X_i}[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)|s_0 = s, a_t \sim \pi].$$

The optimal policy under X_i is

$$\pi^*(X_i) = \arg\max_{\pi} V^{\pi}_{X_i}.$$

Defensive Agent Under Attack: The agent does not observe X directly. Instead, it observes $Y \sim \Pr(Y|X)$. We assume the agent is aware of the attack mechanism and knows the joint distribution $\Pr(X, Y)$, and therefore can compute the posterior $\Pr(X|Y)$.

Given an observation Y, the agent selects a policy $\pi(\cdot|Y)$ that is optimal under posterior uncertainty. We model this as minimizing expected regret:

$$\pi^*(\cdot|Y) = \arg\min_{\pi} \mathbb{E}_{\Pr(X|Y)}\Big[\|V^{\pi}_X - V^{\pi^*(X)}\|_\infty\Big].$$

The performance loss induced by the attack is measured by the expected regret:

$$\mathcal{R} = \mathbb{E}_{\Pr(X,Y)}\Big[\|V^{\pi^*(X)}_X - V^{\pi^*(\cdot|Y)}_X\|_\infty\Big]. \tag{1}$$

Attack Objective: The attacker chooses the stochastic mapping $\Pr(Y|X)$, equivalently the joint distribution $\Pr(X, Y)$, subject to a cost constraint. Let $C(X, Y)$ denote the cost of perturbing X into Y, and let B be the attack budget:

$$\mathbb{E}_{\Pr(X,Y)}\Big[C(X, Y)\Big] \le B.$$

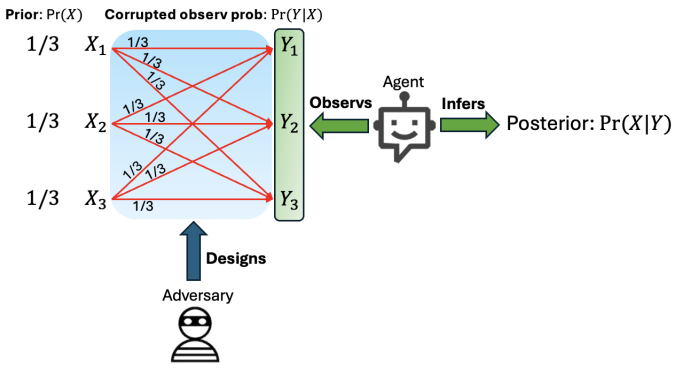


Figure 1: Rate-distortion Attacks on Transition Kernels

We propose a rate-distortion attack, where the attacker solves:

$$\min_{\Pr(X,Y)} I(X;Y), \tag{2}$$

$$\text{subject to } \mathbb{E}_{\Pr(X,Y)}\Big[C(X,Y)\Big] \leq B. \tag{3}$$

Figure 1 illustrates our idea of the rate-distortion attack on the state transition function. Minimizing $I(X;Y)$ forces the perceived variable Y to reveal as little information as possible about the true environment X, subject to a distortion constraint. This creates maximal ambiguity for the agent and induces unavoidable regret, even when the agent is fully aware of the attack and acts optimally under posterior uncertainty.

We discuss the feasibility of performing rate-distortion attacks on transition kernels in detail in the next subsection. In fact, the practical rate-distortion attack on transition functions can be easily implemented by perturbing the agent's state observations randomly.

2.2 Feasibility of Attacking Transition Kernels

We describe how an attacker can implement rate-distortion attacks on the transition kernel during training. First, the environment samples a ground-truth transition kernel X. Before training begins, the attacker generates a false kernel $Y \sim \Pr(\mathrm{Y}|\mathrm{X})$.

This can be achieved in several ways:

- Modify environment parameters that control the dynamics Xu et al. (2023)
- Directly replace X with a constructed kernel Y Rakhsha et al. (2020)
- Manipulate the agent's state observations

In all cases, the agent is effectively trained under the fake kernel Y. In model-based learning, the agent plans using Y and computes a corresponding policy $\pi^*(\cdot|Y)$. In model-free learning, the agent directly learns a policy $\pi^*(Y)$ through interaction, which is optimal for the poisoned environment but not for the true kernel X.

Among all approaches, state-observation attacks are the most practical, as they do not require direct access to or knowledge of the transition kernel. In particular, a simple yet powerful instantiation is a **permutation-based attack**:

Consider a discrete state space $\mathcal{S}$ with $|\mathcal{S}| = S$. The attacker samples a permutation σ uniformly at random from the $S!$ possible permutations and applies it consistently throughout training. That is, whenever the true trajectory is $(s_1, a_1, r_1, s_2, a_2, \ldots, s_{T+1})$, the agent instead observes $\sigma(s_1), a_1, r_1, \sigma(s_2), a_2, \ldots, \sigma(s_{T+1})$. This induces a permuted transition kernel Y that is indistinguishable from a valid environment from the agent's perspective. The randomness in the attack arises from the random choice of permutation σ. Importantly, this attack requires neither modifying the true dynamics nor estimating X; It only alters observations.

In the next section, we present an information-theoretic lower bound on the regret $\mathcal{R}$ incurred by the rate-distortion attacks. For theoretical proofs, we focus on using planning algorithms to find the optimal policy $\pi^*(\cdot|Y)$.

3 Provable Regret Lower Bound for Rate-distortion Attacks

To establish this lower bound, we first introduce the notion of an optimal decoder, one that minimizes the error probability when decoding for X from the delusional observation Y.

Definition:(Optimal decoder) Let $f : \Omega(Y) \to \Omega(X)$ be a decoder function that maps an observation Y to an estimate $\hat{X} = f(Y)$ of the ground-truth transition variable X. The optimal decoder g is defined as the function that minimizes the probability of decoding error

$$g = \arg\min_f \mathbb{E}_{\Pr(X,Y)} \mathbb{1}_{X \neq f(Y)}.$$

We denote the resulting minimum error probability by

$$\mathrm{Pe} = \mathbb{E}_{\Pr(X,Y)} \mathbb{1}_{X \neq g(Y)}.$$

Then we obtain the following lower bound on the regret based on the minimum error probability Pe.

Theorem 3.1. *Assume that for each ground-truth X, the victim agent has a different optimal policy $\pi^*(X)$. Since $\pi^*(X)$ is the optimal policy, there exists a constant $\epsilon > 0$ such that $\|V_X^{\pi^*(X)} - V_X^{\pi}\|_\infty \geq \epsilon$, $\forall \pi \neq \pi^*(X)$. Then, the regret under the attack characterized by $\Pr(X, Y)$ satisfies*

$$\mathcal{R} \geq \epsilon \, \mathrm{Pe},$$

where Pe *is the error probability of the optimal decoder g.*

Moreover,

$$H(\mathrm{Pe}) + \mathrm{Pe} \log |\Omega(X)| \geq H(X|Y) = H(X) - I(X;Y),$$

where $H(\cdot)$ is the entropy.

Proof. We construct an estimator $h : \Omega(Y) \to \Omega(X)$:

$$h(Y) = \begin{cases} \hat{X}, & \text{If } \exists\ \hat{X} \in \Omega(X) \text{ such that } \pi^*(\hat{X}) = \pi^*(\cdot|Y) \\ \arg\min_{\hat{X} \in \Omega(X)} \|V^{\pi^*(\hat{X})} - V^{\pi^*(\cdot|Y)}\|_\infty, & \text{Otherwise.} \end{cases}$$

Assume that there is a bijective relation between the ground-truth transition kernel X and the optimal policy $\pi^*(X)$, we can claim that if $\hat{X} \neq X$, then the optimal policy for the ground-truth X is different from the optimal policy $\pi^*(\cdot|Y)$, namely $\pi^*(X) \neq \pi^*(\cdot|Y)$. Therefore,

$$\mathbb{1}_{X \neq \hat{X}} \leq \mathbb{1}_{\pi^*(X) \neq \pi^*(\cdot|Y)},$$

where $\mathbb{1}$ is the indicator function.

Then we can give a lower bound of the regret R as follows:

$$\begin{aligned} \mathcal{R} &= \mathbb{E}_{\Pr(X,Y)} \|V^{\pi^*(X)} - V^{\pi^*(\cdot|Y)}\|_\infty \\ &\geq \mathbb{E}_{\Pr(X,Y)} [\mathbb{1}_{\pi^*(X) \neq \pi^*(\cdot|Y)} \epsilon] \\ &= \epsilon \mathbb{E}_{\Pr(X,Y)} [\mathbb{1}_{\pi^*(X) \neq \pi^*(\cdot|Y)}] \\ &\geq \epsilon \mathbb{E}_{\Pr(X,Y)} [\mathbb{1}_{X \neq \hat{X}}] \\ &= \epsilon \mathbb{E}_{\Pr(X,Y)} [\mathbb{1}_{X \neq h(Y)}] \\ &\geq \epsilon \mathbb{E}_{\Pr(X,Y)} [\mathbb{1}_{X \neq g(Y)}] \\ &= \epsilon \, \mathrm{Pe}, \end{aligned}$$

where $g(\cdot)$ is the optimal decoder.

Furthermore, we use Fano's inequality Cover & Thomas (2006) from information theory to specify a lower bound on Pe.

Theorem 3.2 (Fano's Inequality)**.** *For any estimator $\hat{X}$ such that $X \to Y \to \hat{X}$, with* $\mathrm{Pe} = \Pr(X \neq \hat{X})$*, we have:*

$$H(\mathrm{Pe}) + \mathrm{Pe} \log |\Omega(X)| \geq H(X|\hat{X}) \geq H(X|Y).$$

This inequality can be weakened to $1 + \mathrm{Pe} \log |\Omega(X)| \geq H(X|Y)$*, or*

$$\mathrm{Pe} \geq \frac{H(X|Y) - 1}{\log |\Omega(X)|}.$$

Using Fano's inequality together with $\mathcal{R} \geq \epsilon \, \mathrm{Pe}$, we conclude the theorem. □

4 Planning under Random Kernels

4.1 Nonexistence of Optimal Policy $\pi^*(\cdot|Y)$ under Random Kernels

In the rate-distortion adversarial attack setting, the agent observes a perturbed transition kernel Y. Conditioned on this observation, the true transition kernel X follows a posterior distribution $\Pr(X|Y)$. Unlike settings where the transition kernel varies across time steps, in our formulation, the kernel is sampled once and remains fixed thereafter.

This distinction introduces new challenges, as the uncertainty is persistent but not dynamically updated. A natural question is whether one can define an optimal policy $\pi^*(\cdot|Y)$ that maximizes value functions under the conditional distribution $\Pr(X|Y)$. One might expect such a policy to maximize the expected value function across all states simultaneously. However, somewhat counterintuitively, we find that such a policy may not exist.

To motivate this, recall that for a standard MDP with a fixed transition kernel P, there always exists an optimal deterministic policy π^* Sutton & Barto (2018). This policy satisfies

$$V^{\pi^*}(s) \geq V^{\pi}(s), \ \forall \pi \neq \pi^*, \forall s \in \mathcal{S}.$$

However, when the transition kernel X is random, a direct extension of this notion becomes problematic. In particular, if we define an optimal deterministic policy π^* as one that maximizes the expected value function at every state, i.e.,

$$\pi^*(s) = \arg\max_{\pi} \mathbb{E}_X[V_X^{\pi}(s)], \forall s \in \mathcal{S}. \tag{4}$$

Then such a policy may not exist.

Theorem 4.1. *Consider an MDP $\{\mathcal{S}, \mathcal{A}, X, r, \gamma\}$, where the transition kernel X is random with distribution $p(X)$. In general, there does not exist a deterministic policy π^* satisfying (4).*

Proof. For a detailed proof, please refer to Appendix A. □

We further investigate whether allowing *stochastic* policies under random kernels resolves the non-existence issue. A stochastic policy is a mapping $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$. One may hope that randomization enables a single policy to maximize the expected value function for all states simultaneously under random kernels. However, this is not generally true. Numerical results under the same setting as Appendix A suggest that there may not exist a stochastic policy satisfying (4).

To illustrate this, we parameterize the policy by $\theta_0 = \pi(a|0)$ and $\theta_1 = \pi(a|1)$. Figure 2 shows contour plots of $\mathbb{E}_X[V_X^{\pi}(0)]$ and $\mathbb{E}_X[V_X^{\pi}(1)]$ under different priors. The maximizing policy parameters (red dots) generally differ across states. For example, under the uniform prior $p = [0.5, 0.5]$, the maximizers are approximately $(1, 0.37)$ for state 0 and $(0, 0)$ for state 1. Hence, no single stochastic policy simultaneously maximizes the expected value for all states. Interestingly, stochastic optimal policies only exist in degenerate cases where the transition kernel becomes deterministic. Additional details are provided in Appendix A.1.

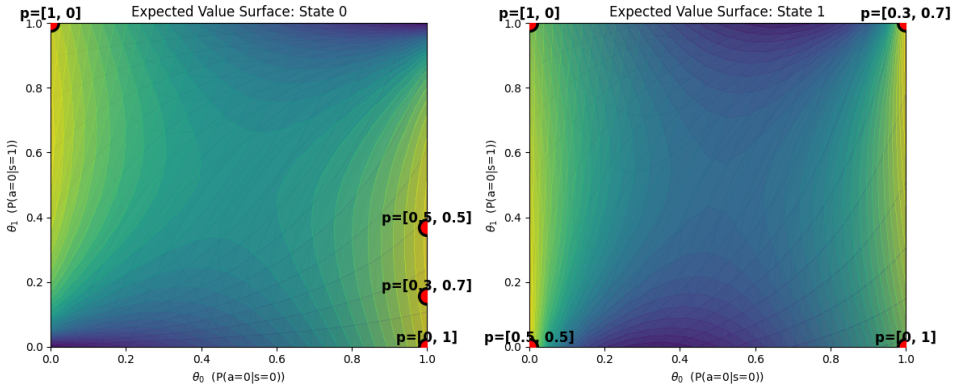


Figure 2: Non-existence of the Optimal Stochastic Policy

4.2 A Natural Extension of Policy Iteration

In a scenario when an optimal deterministic policy π^* defined as (4) exists (We proved in Section 4.1, π^* does not always exist), one might wonder if a naturally extended policy iteration (See 1&2 in Appendix C) under random kernels might converge to this optimal policy.

For a naturally extended policy iteration algorithm, in each iteration, the algorithm evaluates whether a local action update can improve the expected value function under the distribution of transition kernels. If such an improvement exists, the policy is updated accordingly. In practice, this algorithm performs well and often finds an optimal or near-optimal policy with high probability.

However, surprisingly, in contrast to the standard policy iteration, which is guaranteed to converge to the optimal policy under a fixed transition kernel, this natural extension 1&2 *does not* have the same convergence guarantee when the transition kernel remains random.

Theorem 4.2. *Consider an MDP $\{\mathcal{S}, \mathcal{A}, X, r, \gamma\}$, with a random transition kernel $X \sim p(X)$. Assume that it has a deterministic optimal policy $\pi^* = \arg\max_\pi \mathbb{E}_X V_X^\pi$. Then, the extended policy iteration algorithm (Algorithms 1 and 2) does not always converge to π^*.*

Proof. For a detailed proof, please see Appendix B. □

5 Rate-distortion State Observation Attacks on Model-free Learning

Previous sections focused on the rate-distortion attacks on transition functions during planning. We now extend this framework to attacks on state observations, which apply to a broad class of settings, including most model-free reinforcement learning algorithms.

When the state space is large or continuous, and the underlying transition dynamics are implicitly deterministic, an attacker can instead manipulate the agent's observations through information-theoretic random perturbations. Specifically, let $\delta \sim \Omega(\delta)$ denote a random perturbation sampled from a designed distribution $p(\delta)$. At the start of each episode, the attacker samples a perturbation δ, which remains fixed throughout that episode.

At each time step, the agent observes a perturbed state given by the true state plus δ. As a result, the agent's observations are systematically biased, while still appearing stochastic due to the randomness in δ across episodes. Based on these corrupted observations, the agent learns a policy that is optimal for the perturbed states, but generally suboptimal for the true environment.

6 NUMERICAL ANALYSIS

6.1 RATE-DISTORTION ATTACKS WITH ZERO INFORMATION

We start with a randomized (rate-distortion) attack that removes all information about the true transition kernel from the agent's observations. The MDP setting follows Theorem 4.2 (details in Appendix B).

We consider two candidate kernels $X \in \{X_1, X_2\}$ with prior $p(X_1) = p(X_2) = 1/2$. The attacker generates a perturbed kernel Y via a channel $P(Y|X)$. For the zero-information attack, $\Pr(Y = X_j | X = X_i) = 1/2, \forall i, j \in \{1, 2\}$, which yields $\Pr(X|Y) = 1/2$ for all (X, Y) and hence $I(X;Y) = 0$

$X \downarrow Y \rightarrow$	X_1	X_2
X_1	$1 - p_1$	p_1
X_2	p_2	$1 - p_2$

Table 1: Parameters in Minimum Mutual Information Attacks

Under this attack, the agent cannot distinguish between X_1 and X_2 and learns a single policy $\pi^*(\cdot|Y)$ that is identical for both observations: $\pi^*(\cdot|Y_1) = \pi^*(\cdot|Y_2) = \{0 : \text{left}, 1 : \text{left}, 2 : \text{stay}\}$. This "intermediate" policy differs from the true optimal policies under fixed kernels:

$$\pi^*(X_1) = \{0 : \text{left}, 1 : \text{right}, 2 : \text{stay}\}, \pi^*(X_2) = \{0 : \text{stay}, 1 : \text{left}, 2 : \text{right}\}.$$

We measure the performance loss by regret $\mathcal{R}$ defined by (1), and obtain $\mathcal{R} = 3.84$, which is 44.3% of the unattacked value $\mathbb{E}_{p(X)} \| V_X^{\pi^*(X)} \|_\infty$.

Budget-constrained Rate-distortion Attacks Next, we optimize randomized attacks under a budget constraint. Let $p_1 = \Pr(Y = X_2 | X = X_1), p_2 = \Pr(Y = X_1 | X = X_2)$, with fixed costs $c_1 = 1.5$ and $c_2 = 2$. The budget is spent on confusing agents to perceive X_1 as X_2 and vice versa. The budget constraint is $0.5 p_1 c_1 + 0.5 p_2 c_2 < B$.

The attacker designs p_1 and p_2 such that the regret is maximized while not exceeding the budget B.

Maximizing regret over (p_1, p_2) shows that regret increases with B and saturates at 44.3% when $B \approx 0.711$, as shown by Figure 3. The zero-information attack emerges as an extreme point of this tradeoff.

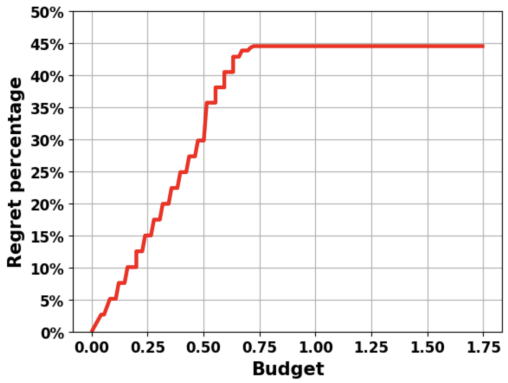


Figure 3: Budget-constrained Rate-distortion Attack: Maximizing Regret

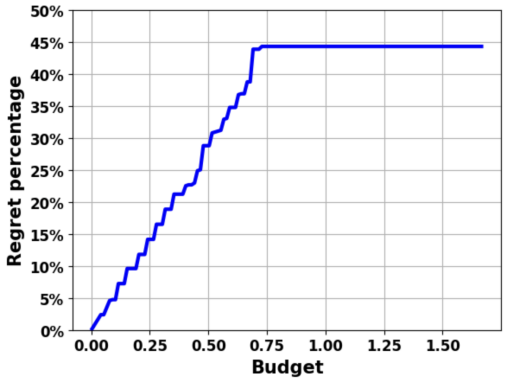


Figure 4: Budget-Constrained Minimized Mutual Information Attack

Minimum Mutual Information Attacks An alternative design principle is to directly minimize $I(X;Y)$ under the same budget using the parametrization in Table 1. Solving

$$\min_{p_1, p_2} I(X;Y)$$
$$\text{s. t. } 0.5 p_1 c_1 + 0.5 p_2 c_2 < B$$

produces attacks that approach the zero-information limit as the budget increases.

As shown in Figure 4, the resulting regret also reaches 44.3% at $B \approx 0.7285$, confirming that the information suppression, rather than a specific attack form, does drive performance degradation.

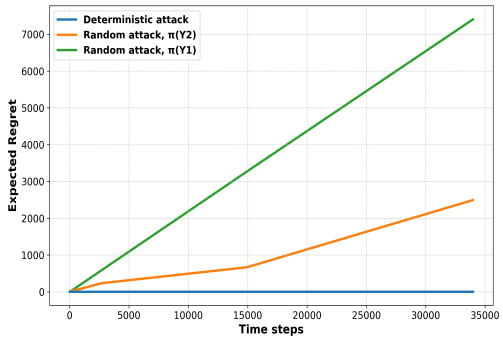


Figure 5: Randomized Attack on Q-learning

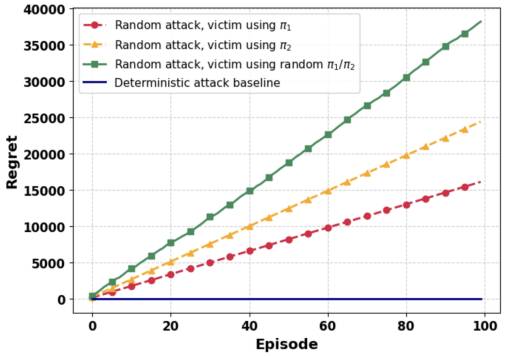


Figure 6: Randomized Attack on DQN

6.2 Rate-distortion Attacks on Reinforcement Learning

The same randomized/rate-distortion attack principle applies beyond planning. We illustrate this idea with both transition function and state-observation perturbations.

Tabular Q-learning (Block-world)Kurtisi (2021) We randomize the environment dynamics through the slip probability α. Let X_1 and X_2 correspond to $\alpha = 0.8$ and $\alpha = 0.2$, with prior $1/2$. Applying the same zero-information channel $(\Pr(X|Y) = 1/2)$ forces the agent to learn under the most ambiguous dynamics. Compared to deterministic attacks that the agent can fully reverse with a 0 regret, the randomized attack induces larger regret, as shown by Figure 5.

Deep Q-learning (Cartpole)Foundation (2025) We implement a ranomized state-observation attack by injecting perturbations δ drawn from a channel $p(\delta_i|\delta_j) = 1/2$, where δ_i is the false noise while δ_j is the true noise. Each perturbation corresponds to a different induced dynamics X_i. As in the planning setting, the agent learns a compromised policy due to ambiguity in observations, leading to significant regret (Figure 6).

Other Variants: Additional implementations, such as random state permutation attacks, further demonstrate that rate-distortion mechanisms can be combined with different attack surfaces. Please refer to Appendix D for more numerical results.

7 Conclusion

We propose a near-optimal type of adversarial attack based on rate-distortion information theory. The attack works by forcing the agent to observe randomized observations, thereby limiting the information it can extract about the true underlying environment. As a result, the agent learns from a systematically distorted view of the environment. We establish a lower bound on the regret induced by the attack and provide numerical results demonstrating its impact on both planning and model-free reinforcement learning. Together, these results show that the proposed attack can significantly reduce the agent's expected state values. We further study the existence of optimal policies and develop a planning framework for MDPs with random transition kernels. Additional discussion on the existence of optimal policies is provided. A limitation is that, when classical optimal policies may not exist under random kernels, new solution concepts are required. Another interesting future direction is to apply rate-distorted state observation attacks on partially-observable MDPs, where the state observations are naturally stochastic.

AI USE STATEMENT

In this work, we used generative AI tools to help generate part of the experiment code. That code was reviewed manually before use. We used the AI tools to help polish the writing of the paper. We have not used generative AI tools to draft or edit the paper, to generate synthetic datasets, to develop the theoretical model, to formulate mathematical claims, to write proofs, to propose the hypotheses tested here, to translate text, to clean datasets, or to carry out qualitative data analysis. Survey design and transcription are not applicable to this work. The numerical results come from our training runs. We take responsibility for the final content of this work.

REFERENCES

Vahid Behzadan and Arslan Munir. Vunerability of deep reinforcement learning to policy induction attacks. In *International Conference on Machine Learning and Data Mining in Pattern Recogonition*, pp. 262–275. Springer, 2017.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, USA, 2006. ISBN 0471241954.

Hamza Fawzi, Paulo Tabuada, and Suhas Diggavi. Secure estimation and control for cyber-physical systems under adversarial attacks. *IEEE Transactions on Automatic Control*, 59(6):1454–1467, 2014. ISSN 1558-2523. doi: 10.1109/tac.2014.2303233. URL http://dx.doi.org/10.1109/TAC.2014.2303233.

Farama Foundation. Cartpole environment documentation. https://gymnasium.farama.org/environments/classic_control/cart_pole/, 2025. Gymnasium documentation.

Tim Franzmeyer, Stephen Marcus McAleer, João F. Henriques, Jakob N. Foerster, Philip H. S. Torr, Adel Bibi, and Christian Schroeder de Witt. Illusory attacks: Information-theoretic detectability matters in adversarial attacks. In *Proceedings of the 12th International Conference on Learning Representations (ICLR 2024)*, 2024. URL https://openreview.net/forum?id=F5dhGCdyYh. Spotlight presentation.

Yunhan Huang and Quanyan Zhu. Deceptive reinforcement learning under adversarial manipulations on cost signals. In *International Conference on Decision and Game Theory for Security*, pp. 217–237. Springer, 2019.

Jihun Kim and Javad Lavaei. Prevailing against adversarial noncentral disturbances: Exact recovery of linear systems with the l¡sub¿1¡/sub¿-norm estimator. In *2025 American Control Conference (ACC)*, pp. 1161–1168. IEEE, 2025. doi: 10.23919/acc63710.2025.11107807. URL http://dx.doi.org/10.23919/ACC63710.2025.11107807.

Senad Kurtisi. Q-learning block world. https://github.com/senadkurtisi/Q-learning-block-world, 2021. Accessed: 2025-05-15.

Michael Littman. Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the Eleventh International Conference*, 1994.

Guanlin Liu and Lifeng Lai. Provably efficient black-box action poisoning attacks against reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 12400–12410, 2021.

Guanlin Liu and Lifeng Lai. Efficient adversarial attacks on online multi-agent reinforcement learning. In *arXiv preprint arXiv:2307.07670*, 2023.

Xiao-Yang Liu, Hongyang Yang, Qian Chen, Runjia Zhang, Liuqing Yang, Bowen Xiao, and Christina Dan Wang. Finrl: A deep reinforcement learning library for automated stock trading in quantitative finance. In *Deep RL Workshop*. NeurIPS, 2020.

Yuzhe Ma, Xuezhou Zhang, Wen Sun, and Jerry Zhu. Policy poisoning in batch reinforcement learning and control. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Nicholas Mastronarde and Mihaela van der Schaar. Fast reinforcement learning for energy-efficient wireless communication. *IEEE Transactions on Signal Processing*, 59(12):6262–6266, 2011. doi: 10.1109/TSP.2011.2165211.

Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning, 2017. URL https://arxiv.org/abs/1703.02702.

Amin Rakhsha, Goran Radanovic, Rati Devidze, Xiaojin Zhu, and Adish Singla. Policy teaching via environment poisoning: Training-time adversarial attacks against reinforcement learning, 2020. URL https://arxiv.org/abs/2003.12909.

Shalev-Shwartz, Shaked Shammah, and Amnon Shuashua. Safe, multi-agent, reinforcement learning for autonomous driving. In *arXiv preprint arXiv:1610.03295*, 2016.

Yunlong Song, Mats Steinweg, Elia Kaufmann, and Davide Scaramuzza. Autonomous drone racing with deep reinforcement learning. In *International Conference on Intelligent Robots and Systems (IROS)*. IEEE/RSJ, 2021.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2 edition, 2018. URL http://incompleteideas.net/book/the-book-2nd.html.

Hang Xu, Xinghua Qu, and Zinovi Rabinovich. Policy resilience to environment poisoning attacks on reinforcement learning. In *arXiv preprint arXiv:2304.12151*, 2023.

Jing Yu, Clement Gehring, Florian Schäfer, and Animashree Anandkumar. Robust reinforcement learning: A constrained game-theoretic approach. In Ali Jadbabaie, John Lygeros, George J. Pappas, Pablo A.nbsp;Parrilo, Benjamin Recht, Claire J. Tomlin, and Melanie N. Zeilinger (eds.), *Proceedings of the 3rd Conference on Learning for Dynamics and Control*, volume 144 of *Proceedings of Machine Learning Research*, pp. 1242–1254. PMLR, 07 – 08 June 2021. URL https://proceedings.mlr.press/v144/yu21a.html.

Huang Zhang, Hongge Chen, Chaowei Xiao, Bo Li, Mingyan Liu, Duane Boning, and Cho-Jui Hsieh. Robust deep reinforcement learning against adversarial perturbations on state observations. In *International Conference on Decision and Game Theory for Security*, 2020a.

Xuezhou Zhang, Yuzhe Ma, Adish Singla, and Xiaojin Zhu. Adaptive reward-poisoning attacks against reinforcement learning. In *International Conference on Machine Learning*, volume 119, pp. 11225–11234, 2020b.

Xiangyu Zhao, Long Xia, Liang Zhang, Zhuoye Ding, and Jiliang Tang. Deep reinforcement learning for page-wise recommendations. In *The 12th ACM Conference on Recommender Systems*, pp. 95–103. ACM, 2018.

A Proof of Theorem 4.1

Settings: Consider a discounted MDP with $\mathcal{S} = \{0, 1\}$, $\mathcal{A} = \{a, b\}$, $\gamma = 0.9$, and a random kernel X with values X_1 and X_2 as follows:

$$X_1(a) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},\ X_1(b) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, X_2(a) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\ X_2(b) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

The notation means, for example, $X_1(a)$ is the sampled state transition function X_1, and the agent takes action a. The prior distribution $p(X_1) = p(X_2) = 0.5$. Table 2 shows the reward function$r(s, s')$.

We use exhaustive search to find the optimal policy π^* in this example. The space of policy is $\Pi = \{\ \pi_1 = \{0 : a, 1 : a\}, \pi_2 = \{0 : a, 1 : b\}, \pi_3 = \{0 : b, 1 : a\}, \pi_4 = \{0 : b, 1 : b\}\}$. To evaluate every policy $\pi \in \Pi$ with a fixed kernel X_i, we apply the Bellman operator:

$$\begin{aligned} V^{\pi}_{X_i}(s) &= \sum_{s'} X_i^{\pi}[r(s, \pi(s), s') + \gamma V^{\pi}(s)] \\ &= \bar{r}^{\pi}(s) + \gamma \sum_{s'} X_i^{\pi} V^{\pi}(s) \end{aligned}$$

In the Bellman equation, the notation X_i^π means $X_i^\pi = X_i(s'|s, \pi(s)), \forall s$. The notation $\bar{r}^\pi(s) = \sum_{s'} X_i^\pi(s'|s, \pi(s)) r(s, \pi(s), s')$ depends only on s, since $\pi(s)$ is fixed.

Therefore, the Bellman equation for $V_{X_i}^\pi$ defines a system of 2 linear equations with 2 variables, and the vector form is

$$V_{X_i}^\pi = \bar{r}^\pi + \gamma X_i^\pi V_{X_i}^\pi.$$

Solving the above equation, we get $V_{X_i}^\pi(s), \forall s$:

$$V_{X_i}^\pi = (I - \gamma X_i^\pi)^{-1} \bar{r}^\pi. \tag{5}$$

By computing the state values for each kernel X_i using equation (5), we further evaluated the expected V-values $\mathbb{E}_X V_X^\pi(s)$ for every state and every policy. We see that from Table 3 for state 0, the optimal policy is $= \pi_2$ *or* π_3, but for state 1, the optimal policy is π_1 *or* π_4. Notice that an optimal policy π^* needs to satisfy $E_X V_X^{\pi^*}(s) > E_X V_X^\pi(s), \forall \pi$ for all states simultaneously. Therefore, this is a counterexample showing that there exists an MDP with a random kernel that does not have an optimal policy. Therefore, we proved Theorem 4.1.

$s \downarrow s' \rightarrow$	0	1
0	0.06	0.15
1	0.3	0.95

Table 2: Reward Function in Thereom 4.1

state $\downarrow$ policy$\rightarrow$	π_1	π_2	π_3	π_4
0	1.41	4.65	4.65	1.41
1	5.89	5.17	5.17	5.89

Table 3: Expected reward over transition kernels: $\mathbb{E}_X V_X^\pi(s)$

A.1 Non-existence of Stochastic Optimal Policy

We further investigate whether a stochastic optimal policy π^* exists under the criterion in (4). A stochastic policy is defined as a mapping $\pi^* : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$.

To study this question, we conduct a numerical experiment under the same setting as in Appendix A. Our goal is to determine whether there exists a single stochastic policy that simultaneously maximizes the expected value function for all states.

The result indicates that such a policy does not always exist. In particular, there may be no policy π^* satisfying

$$\pi^* = \arg\max_\pi \mathbb{E}_X[V_X^\pi(s)], \forall s \in \mathcal{S}.$$

To illustrate this, we parametrize the policy using two variables: $\theta_0 = \pi(a|0)$ and $\theta_1 = \pi(a|1)$, representing the probabilities of taking action a in states 0 and 1, respectively. The probabilities of taking action b are then $1 - \theta_0$ and $1 - \theta_1$.

Figure 2 presents contour plots of the the expected state values $\mathbb{E}_X[V_X^\pi(0)]$ and $\mathbb{E}_X[V_X^\pi(1)]$ as functions of (θ_0, θ_1). Lighter colors indicate higher expected values. We consider four different priors over the transition kernel $p = \{[0, 1], [1, 0], [0.3, 0.7], [0.5, 0.5]\}$, and mark the optimal policies for each state with red markers. The contour plots correspond to the case corresponding to the uniform prior $p = [0.5, 0.5]$.

Under this prior, the maximizing policies differ across states: The optimal parameters are approximately $(\theta_0, \theta_1) = (1, 0.37)$ for state 0 and $(\theta_0, \theta_1) = (0, 0)$ for state 1. Since these optima do not coincide, there is no single policy that maximizes the expected value for both states simultaneously.

For a stochastic optimal policy to exist under (4), the same policy parameters must maximize the expected value for all states, i.e., the red markers in both subplots for a given prior must coincide. This condition fails in cases when priors are $[0.3, 0.7]$ or $[0.5, 0.5]$.

Finally, we observe that a stochastic optimal policy exists only in the degenerate cases $p = [0, 1]$ and $p = [1, 0]$, where the transition kernel is deterministic.

B PROOF OF THEOREM 4.2

Consider a 3-state, 3-action discounted MDP. The state space is $\mathcal{S} = \{0, 1, 2\}$, and the states are connected in a circle. The action space is $\mathcal{A} = \{\text{left}, \text{right}, \text{stay}\}$.and the discount factor $\gamma = 0.9$. The ground-truth transition probability X takes two possible values $\{X_1, X_2\}$ with the prior distribution $p(X_1) = p(X_2) = 0.5$. Tables 5 and 6 give the transition kernels X_1 and X_2 with respect to every action in $\mathcal{A}$. The reward function $r(s, s')$ is displayed in Table 4:

$s \downarrow \ s' \rightarrow$	0	1	2
0	0.06	0.1	0.15
1	0.02	0.1	0.15
2	0.01	0.2	0.95

Table 4: Reward Function in Theorem 4.2

0	0	1
1	0	0
0	1	0

0	1	0
0	0	1
1	0	0

1	0	0
0	1	0
0	0	1

Table 5: $X_1(\text{left}),\ X_1(\text{right}),\ X_1(\text{stay})$

0	1	0
0	0	1
1	0	0

1	0	0
0	1	0
0	0	1

0	0	1
1	0	0
0	1	0

Table 6: $X_2(\text{left}),\ X_2(\text{right}),\ X_2(\text{stay})$

The optimal solution under the above setting is $\pi^*(\cdot|Y) = \{0 : \text{left}, 1 : \text{left}, 2 : \text{stay}\}$, found by exhaustive search. The optimal solution under fixed transition kernel X_1 is $\{0 : \text{left}, 1 : \text{right}, 2 : \text{stay}\}$, and the optimal solution under fixed kernel X_2 is $\{0 : \text{stay}, 1 : \text{left}, 2 : \text{right}\}$. Notice that $\pi^*(\cdot|Y) \neq \pi^*(X_1)$, and $\pi^*(\cdot|Y) \neq \pi^*(X_2)$.

Notice that in our setting, the three probability kernels in X_2 associated with actions are the permutation of probability kernels in X_1, i.e., $X_2(\text{left}) = X_1(\text{right})$, $X_2(\text{right}) = X_1(\text{stay})$, and $X_2^{\text{stay}} = X_1^{\text{left}}$. Let $r_{X_i}(s, a) = \sum_{s'} X_i(s'|s, a) r(s, a, s')$ denote the immediate reward gained under the fixed X_i. We have that the largest immediate reward gained under X_1 is the same as the largest immediate reward gained under X_2, $\max_{a \in \mathcal{A}} r_{X_1}(s, a) = \max_{a \in \mathcal{A}} r_{X_2}(s, a), \forall s$.

We claim that with the prior probability $p(X_1) = p(X_2) = 1/2$, and with the initial policy $\pi_0 = \pi^*(X_1)$, Algorithm 2 fails to converge to π^*.

Starting with $\pi_0 = \pi^*(X_1)$, then by policy evaluation, we have that the updated V-values with fixed kernel X_1 by following π_0 is $V_{X_1}^{\pi_0} = \{0 : 7.858, 1 : 7.858, 2 : 8.658\}$, i.e., $V_{X_1}^{\pi_0}(0) = 7.858$, $V_{X_1}^{\pi_0}(1) = 7.858$ and $V_{X_1}^{\pi_0}(2) = 8.658$. The V-value with fixed kernel X_2 following π_0 is updated to $V_{X_2}^{\pi_0} = \{0 : 0.911, 1 : 0.911, 2 : 1.02\}$. Notice that $V_{X_1}^{\pi_0}(s) \geq V_{X_2}^{\pi_0}(s), \forall s$. The extended policy iteration updates $\pi_1(s)$ such that:

$$
\begin{aligned}
\pi_1(s) = \arg\max_a \ & 1/2 \sum_{s'} X_1(s'|s, a)[r(s, a, s') + \gamma V_{X_1}^{\pi_0}(s')] \\
& + 1/2 \sum_{s'} X_2(s'|s, a)[r(s, a, s') + \gamma V_{X_2}^{\pi_0}(s')] \\
= \arg\max_a \ & 1/2 (r_{X_1}(s, a) + \gamma \sum_{s'} X_1(s'|s, a) V_{X_1}^{\pi_0}(s')) \\
& + 1/2 (r_{X_2}(s, a) + \gamma \sum_{s'} X_2(s'|s, a) V_{X_2}^{\pi_0}(s')) \\
= \arg\max_a \ & 1/2 Q_{X_1}^{\pi_0}(s, a) + 1/2 Q_{X_2}^{\pi_0}(s, a)
\end{aligned}
$$

The extended Q-values $Q^{\pi_0}(s, a) = 1/2 Q_{X_1}^{\pi_0}(s, a) + 1/2 Q_2^{\pi_0}(s, a)$ are shown in Table 7. Since for all states $\{0, 1, 2\}$, we have that $\pi_1(0) = \arg\max_a Q^{\pi_0}(0, a) = 0, \pi_1(1) = \arg\max_a Q^{\pi_0}(1, a) = 1$, and $\pi_1(2) = \arg\max_a Q^{\pi_0}(2, a) = 2$. The updated policy has no change: $\pi_1 = \pi_0 = \pi^*(X_1)$. Therefore, the evaluations of $V_{X_i}^{\pi_1}(s)$ stay the same as $V_{X_i}^{\pi_0}(s), \forall s$. The update of the following sequence of policies $\{\pi_2, \pi_3, \ldots, \pi_n\}$ repeats the above process, and are all equal to the initial policy, i.e. $\pi_n = \pi_0 = \pi^*(X_1)$. Therefore, the algorithm 1&2 fails to converge to $\pi^*(\cdot|Y)$.

$s \downarrow a \rightarrow$	left	right	stay
0	4.43	4.02	4.10
1	4.08	4.43	4.01
2	4.05	4.47	4.88

Table 7: $Q^{\pi_0}(s, a)$

C Extended Policy Iteration Algorithm under Random Kernels

Algorithm 1 Policy Evaluation

Input: $\mathcal{S}, \mathcal{A}, \Omega(X), r, \gamma, \pi$

for $X_i \in \Omega(X)$ **do**
  Repeat
  $\Delta_{X_i} \leftarrow 0$
  for $s \in \mathcal{S}$ **do**
    $v_{X_i} \leftarrow V^{\pi}_{X_i}(s)$
    $V^{\pi}_{X_i}(s) \leftarrow \sum_{s'} X_i(s'|s, \pi(s))[r(s, \pi(s), s') + \gamma V^{\pi}_{X_i}(s')]$
    $\Delta_{X_i} \leftarrow \max(v_{X_i} - V^{\pi}_{X_i}(s))$
  end for
  until $\Delta_{X_i} < \theta$ ▷ θ is a small threshold
end for
return $V^{\pi}_{X_i}, X_i \in \Omega(X_i)$

Algorithm 2 Policy Iteration under Random Kernels

Input: $\mathcal{S}, \mathcal{A}, V^{\pi}_{X_i}, \Omega(X), r, \gamma, \pi, p, n$

for k from 1 to $n-1$ **do**
  for $s \in \mathcal{S}$, **do**
    $\pi_{k+1}(s) = \arg_a \sum_{X_i} p(X_i)(\sum_{s'} X_i(s'|s, a) \cdot$
      $[r(s, a, s') + \gamma V^{\pi_k}_{X_i}(s')])$
  end for
  $V^{k+1}_{X_i}$ =PolicyEvaluation(π_{k+1})
end for
return π_n

D Extra Numerical Results: Randomized State-observation Attacks via Random Permutations

We demonstrate that the proposed information-theoretic rate-distortion attack can be implemented through randomized perturbation on state observations during training.

In this experiment, the environment contains 3 states and 3 actions. For each action, the dynamics are represented by a 3×3 transition matrix. There are 6 possible transition kernels $\{X_1, X_2, \ldots, X_6\}$. One representative kernel, X_1, is shown in Table 9. The remaining 5 kernels are obtained by permuting the rows and columns of X_1, that is,

$$X_i = P_i X_1 P_i^T, i \in \{1, 2, \ldots, 6\},$$

where P_i is a permutation matrix. We assume a uniform prior over the 6 kernels. The reward function $r(s, a)$ is given in Table 8.

Since the state spaces contain three states, there are $3! = 6$ possible state permutations. Each permutation corresponds to simultaneously permuting the rows and columns of the transition matrices. For example, mapping state 0 to state 1 while leaving the other states unchanged is equivalent to swapping the first and second rows and columns of the kernel.

$s \downarrow a \rightarrow$	left	right	stay
0	1.0	1.3	1.1
1	2.0	2.3	2.1
2	3.0	3.3	3.1

Table 8: Reward Function in Randomized Permutation Attack

1	0	0	0.2	0.8	0	0.9	0.1	0
0.8	0.2	0	0	0.2	0.8	0.1	0.8	0.1
0	0.8	0.2	0	0	1	0	0.1	0.9

Table 9: $X_1(\text{left}),\ X_1(\text{right}),\ X_1(\text{stay})$

During each training episode, the environment first samples a ground-truth transition kernel from the prior distribution. The agent then collects transition samples

$$(s_1, a_1, r_1, s_2), (s_2, a_2, r_2, s_3), \ldots, (s_T, a_T, r_T, s_{T+1}),$$

with $T = 100$, and estimates the transition kernel empirically.

The attacker independently samples one of the six state permutations in each episode and applies it consistently throughout trajectory collection. Consequently, the agent observes permuted states rather than the true states, inducing a randomized delusional transition model. Figure 7 shows the resulting increase in regret over 20 training episodes.

E Technical Details for Reproducing Numerical Results

Computing infrastructure: For all numerical experiments, we used 32 NVIDIA GeForce RTX 2080 Ti GPUs. The memory limit is $32GB$.
Code Implementation: We provide our code implementations of the three experiments on rate-distortion adversarial attacks.

Budget-constrained Rate-distortion Attack:

`https://drive.google.com/file/d/1G7gpTC7ZT9tnWhuZEpmbjDuxb7jvRVyb/view?usp=share_link`

Rate-distortion Attack on Q-learning:

`https://drive.google.com/file/d/1PUpObiQT_TaJC9Trz1wfyj4WPsxZ2Xwe/view?usp=share_link`

Rate-distortion Attack on DQN:

`https://drive.google.com/file/d/13jT8a0922RKtkO-YqsDydIZSl4xOAQWV/view?usp=share_link`

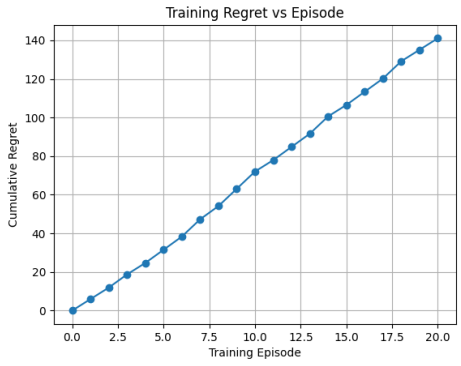


Figure 7: Regret under Rate-distortion State Observation Attacks