Reward-Poisoning Attacks on Offline Multi-Agent Reinforcement Learning

Young Wu, Jeremy McMahan, Xiaojin Zhu, Qiaomin Xie

University of Wisconsin-Madison
yw@cs.wisc.edu, jmcmahan@wisc.edu, jerryzhu@cs.wisc.edu, qiaomin.xie@wisc.edu

Abstract

In offline multi-agent reinforcement learning (MARL), agents estimate policies from a given dataset. We study reward-poisoning attacks in this setting where an exogenous attacker modifies the rewards in the dataset before the agents see the dataset. The attacker wants to guide each agent into a nefarious target policy while minimizing the L^p norm of the reward modification. Unlike attacks on single-agent RL, we show that the attacker can install the target policy as a Markov Perfect Dominant Strategy Equilibrium (MPDSE), which rational agents are guaranteed to follow. This attack can be significantly cheaper than separate single-agent attacks. We show that the attack works on various MARL agents including uncertainty-aware learners, and we exhibit linear programs to efficiently solve the attack problem. We also study the relationship between the structure of the datasets and the minimal attack cost. Our work paves the way for studying defense in offline MARL.

Introduction

Multi-agent reinforcement learning (MARL) has achieved tremendous empirical success across a variety of tasks such as autonomous driving, cooperative robotics, economic policy-making, and video games. In MARL, several agents interact with each other and the underlying environment, and each of them aims to optimize their individual long-term reward (Zhang, Yang, and Başar 2021). Such problems are often formulated under the framework of Markov Games (Shapley 1953), which generalizes the Markov Decision Process model from single-agent RL. In offline MARL, the agents aim to learn a good policy by exploiting a pre-collected dataset without further interactions with the environment or other agents (Pan et al. 2022; Jiang and Lu 2021; Cui and Du 2022; Zhong et al. 2022). The optimal solution in MARL typically involves equilibria concepts.

While the above empirical success is encouraging, MARL algorithms are susceptible to data poisoning attacks: the agents can reach the wrong equilibria if an exogenous attacker manipulates the feedback to agents. For example, a third-party attacker may want to interfere with traffic to cause autonomous vehicles to behave abnormally; teach robots an incorrect procedure so that they fail at certain tasks; misinform economic agents about the state of the economy and guide them to make irrational investments or saving decisions; or cause the non-player characters in a video game to behave improperly to benefit certain human players. In this paper, we study the security threat posed by reward-poisoning attacks on offline MARL. Here, the attacker wants the agents to learn a target policy $\pi^\dagger$ of the attacker's choosing ($\pi^\dagger$ does not need to be an equilibrium in the original Markov Game). Meanwhile, the attacker wants to minimize the amount of dataset manipulation to avoid detection and accruing high cost. This paper studies optimal offline MARL reward-poisoning attacks. Our work serves as a first step toward eventual defense against reward-poisoning attacks.

Our Contributions

We introduce reward-poisoning attacks in offline MARL. We show that any attack that reduces to attacking single-agent RL separately must be suboptimal. Consequently, new innovations are necessary to attack effectively. We present a reward-poisoning framework that guarantees the target policy $\pi^\dagger$ becomes a Markov Perfect Dominant Strategy Equilibrium (MPDSE) for the underlying Markov Game. Since any rational agent will follow an MPDSE if it exists, this ensures the agents adopt the target policy $\pi^\dagger$. We also show the attack can be efficiently constructed using a linear program.

The attack framework has several important features. First, it is effective against a large class of offline MARL learners rather than a specific learning algorithm. Second, the framework allows partially decentralized agents who can only access their own individual rewards rather than the joint reward vectors of all agents. Lastly, the framework only makes the minimal assumption on the rationality of the learners that they will not take dominated actions.

We also give interpretable bounds on the minimal cost to poison an arbitrary dataset. These bounds relate the minimal attack cost to the structure of the underlying Markov Game. Using these bounds, we derive classes of games that are especially cheap or expensive for the attacker to poison. These results show which games may be more susceptible to an attacker, while also giving insight to the structure of multi-agent attacks.

In the right hands, our framework could be used by a benevolent entity to coordinate agents in a way that im-

proves social welfare. However, a malicious attacker could exploit the framework to harm learners and only benefit themselves. Consequently, our work paves the way for future study of MARL defense algorithms.

Related Work

Online Reward-Poisoning: Reward poisoning problem has been studied in various settings, including online single-agent reinforcement learners (Banihashem et al. 2022; Huang and Zhu 2019; Liu and Lai 2021; Rakhsha et al. 2021a,b, 2020; Sun, Huo, and Huang 2020; Zhang et al. 2020), as well as online bandits (Bogunovic et al. 2021; Garcelon et al. 2020; Guan et al. 2020; Jun et al. 2018; Liu and Shroff 2019; Lu, Wang, and Zhang 2021; Ma et al. 2018; Yang et al. 2021; Zuo 2020). Online reward poisoning for multiple learners is recently studied as a game redesign problem in (Ma, Wu, and Zhu 2021).

Offline Reward Poisoning: Ma et al. (2019); Rakhsha et al. (2020, 2021a); Rangi et al. (2022b); Zhang and Parkes (2008); Zhang, Parkes, and Chen (2009) focus on adversarial attack on offline single-agent reinforcement learners. Gleave et al. (2019); Guo et al. (2021) study the poisoning attack on multi-agent reinforcement learners, assuming that the attacker controls one of the learners. Our model instead assumes that the attacker is not one of the learners, and the attacker wants to and is able to poison the rewards of all learners at the same time. Our model pertains to many applications such as autonomous driving, robotics, traffic control, and economic analysis, in which there is a central controller whose interests are not aligned with any of the agents and can modify the rewards and therefore manipulate all agents at the same time.

Constrained Mechanism Design: Our paper is also related to the mechanism design literature, in particular, the K-implementation problem in (Monderer and Tennenholtz 2004; Anderson, Shoham, and Altman 2010). Our model differs mainly in that the attacker, unlike a mechanism designer, does not alter the game/environment directly, but instead modifies the training data, from which the learners infer the underlying game and compute their policy accordingly. In practical applications, rewards are often stochastic due to imprecise measurement and state observation, hence the mechanism design approach is not directly applicable to MARL reward poisoning. Conversely, constrained mechanism design can be viewed as a special case when the rewards are deterministic and the training data has uniform coverage of all period-state-action tuples.

Defense against Attacks on Reinforcement Learning: There is also recent work on defending against reward poisoning or adversarial attacks on reinforcement learning; examples include (Banihashem, Singla, and Radanovic 2021; Lykouris et al. 2021; Rangi et al. 2022a; Wei, Dann, and Zimmert 2022; Wu et al. 2022; Zhang et al. 2021a,b). These work focus on the single-agent setting where attackers have limited ability to modify the training data. We are not aware of defenses against reward poisoning in our offline multi-agent setting. Given the numerous real-world applications of offline MARL, we believe it is important to study the multi-agent version of the problem.

Preliminaries

Markov Games. A finite-horizon general-sum n-player Markov Game is given by a tuple $G = (\mathcal{S}, \mathcal{A}, P, R, H, \mu)$ (Littman 1994). Here $\mathcal{S}$ is the finite state space, and $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$ is the finite joint action space. We use $\boldsymbol{a} = (a_1, \ldots, a_n) \in \mathcal{A}$ to represent a joint action of the n learners; we sometimes write $\boldsymbol{a} = (a_i, a_{-i})$ to emphasize that learner i takes action a_i and the other $n-1$ learners take joint action a_{-i}. For each period $h \in [H]$, $P_h : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition function, where $\Delta(\mathcal{S})$ denotes the probability simplex on $\mathcal{S}$, and $P_h(s'|s, \boldsymbol{a})$ is the probability that the state is s' in period $h+1$ given the state is s and the joint action is $\boldsymbol{a}$ in period h. $\boldsymbol{R}_h : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^n$ is the mean reward function for the n players, where $R_{i,h}(s, \boldsymbol{a})$ denotes the scalar mean reward for player i in state s and period h when the joint action $\boldsymbol{a}$ is taken. The initial state distribution is μ.

Policies and value functions. We use π to denote a deterministic Markovian *policy* for the n players, where $\boldsymbol{\pi}_h : \mathcal{S} \to \mathcal{A}$ is the policy in period h and $\boldsymbol{\pi}_h(s)$ specifies the joint action in state s and period h. We write $\boldsymbol{\pi}_h = (\pi_{i,h}, \pi_{-i,h})$, where $\pi_{i,h}(s)$ is the action taken by learner i and $\pi_{-i,h}(s)$ is the joint action taken by learners other than i in state s period h. The *value* of a policy π represents the expected cumulative rewards of the game assuming learners take actions according to π. Formally, the Q value of learner i in state s in period h under a joint action $\boldsymbol{a}$ is given recursively by

$$Q^{\pi}_{i,H}(s, \boldsymbol{a}) = R_{i,H}(s, \boldsymbol{a}),$$
$$Q^{\pi}_{i,h}(s, \boldsymbol{a}) = R_{i,h}(s, \boldsymbol{a}) + \sum_{s' \in \mathcal{S}} P_h(s'|s, \boldsymbol{a}) V^{\pi}_{i,h+1}(s').$$

The value of learner i in state s in period h under policy π is given by $V^{\pi}_{i,h}(s) = Q^{\pi}_{i,h}(s, \boldsymbol{\pi}_h(s))$, and we use $\boldsymbol{V}^{\pi}_h(s) \in \mathbb{R}^n$ to denote the vector of values for all learners in state s in period h under policy π.

Offline MARL. In offline MARL, the learners are given a *fixed* batch dataset $\mathcal{D}$ that records historical plays of n agents under some behavior policies, and no further sampling is allowed. We assume that $\mathcal{D} = \left\{\left(s^{(k)}_h, \boldsymbol{a}^{(k)}_h, \boldsymbol{r}^{0,(k)}_h\right)_{h=1}^{H}\right\}_{k=1}^{K}$ contains K episodes of length H. The data tuple in period h of episode k consists of the state $s^{(k)}_h \in \mathcal{S}$, the joint action profile $\boldsymbol{a}^{(k)}_h \in \mathcal{A}$, and reward vector $\boldsymbol{r}^{0,(k)}_h \in \mathbb{R}^n$, where the superscript 0 denotes the original rewards before any attack. The next state $s^{(k)}_{h+1}$ can be found in the next tuple. Given the shared data $\mathcal{D}$, each learner independently constructs a policy π_i to maximize their own cumulative reward. They then behave according to the resulting joint policy $\pi = (\pi_1, \ldots, \pi_n)$ in future deployment. Note that in a multi-agent setting, the learners' optimal solution concept is typically an approximate Nash equilibrium or Dominant Strategy Equilibrium (Cui and Du 2022; Zhong et al. 2022).

An agent's access to $\mathcal{D}$ may be limited, for example, due to privacy reasons. There are multiple levels of accessibility. In the first level, the agents can only access data that directly involves itself: instead of the tuple $(s_h, \boldsymbol{a}_h, \boldsymbol{r}_h)$, agent i would only be able to see $(s_h, a_{i,h}, r_{i,h})$. In the second level, agent i can see the joint action but only its own reward: $(s_h, \boldsymbol{a}_h, r_{i,h})$. In the third level, agent i can see the whole $(s_h, \boldsymbol{a}_h, \boldsymbol{r}_h)$. We focus on the second level in this paper.

Let $N_h(s, \boldsymbol{a}) = \sum_{k=1}^{K} \mathbf{1}_{\{s_h^{(k)}=s, \boldsymbol{a}_h^{(k)}=\boldsymbol{a}\}}$ be the total number of episodes containing $(s, \boldsymbol{a}, \cdot)$ in period h. We consider a dataset $\mathcal{D}$ that satisfies the following coverage assumption.

Assumption 1. (Full Coverage) For each $(s, \boldsymbol{a})$ and h, $N_h(s, \boldsymbol{a}) > 0$.

While this assumption might appear strong, we later show that it is necessary to effectively poison the dataset.

Attack Model

We assume that the attacker has access to the original dataset $\mathcal{D}$. The attacker has a pre-specified target policy $\boldsymbol{\pi}^\dagger$ and attempts to poison the rewards in $\mathcal{D}$ with the goal of forcing the learners to learn $\boldsymbol{\pi}^\dagger$ from the poisoned dataset. The attacker also desires that the attack has a minimal cost. We let $C(r^0, r^\dagger)$ denote the cost of a specific poisoning, where $r^0 = \left\{\left(\boldsymbol{r}_h^{0,(k)}\right)_{h=1}^{H}\right\}_{k=1}^{K}$ are the original rewards and $r^\dagger = \left\{\left(\boldsymbol{r}_h^{\dagger,(k)}\right)_{h=1}^{H}\right\}_{k=1}^{K}$ are the poisoned rewards. We focus on the L^1-norm cost $C(r^0, r^\dagger) = \|r^0 - r^\dagger\|_1$.

Rationality. For generality, the attacker makes minimal assumptions about the learners' rationality. Namely, the attacker only assumes that the learners never take dominated actions (Monderer and Tennenholtz 2004). For technical reasons, we strengthen this assumption slightly by introducing an arbitrarily small margin $\iota > 0$ (e.g. representing the learners' numerical resolution).

Definition 1. A ι-strict Markov perfect dominant strategy equilibrium (ι-MPDSE) of a Markov Game G is a policy π satisfying that for all learners $i \in [n]$, periods $h \in [H]$, and states $s \in \mathcal{S}$,

$$\forall\, a_i \in \mathcal{A}_i, a_i \neq \pi_{i,h}(s), a_{-i} \in \mathcal{A}_{-i}: \\ Q_{i,h}^{\pi}(s, (\pi_{i,h}(s), a_{-i})) \geq Q_{i,h}^{\pi}(s, (a_i, a_{-i})) + \iota.$$

Note that a strict MPDSE, if exists, must be unique.

Assumption 2. (Rationality) The learners will play an ι-MPDSE should one exist.

Uncertainty-aware attack. State-of-the-art MARL algorithms are typically uncertainty-aware (Cui and Du 2022; Zhong et al. 2022), meaning that learners are cognizant of the model uncertainty due to finite, random data and will calibrate their learning procedure accordingly. The attacker accounts for such uncertainty-aware learners but does not know the learners' specific algorithm or internal parameters. It only assumes that the policies computed by the learners are solutions to some game that is plausible given the dataset. Accordingly, the attacker aims to poison the dataset in such a way that the target policy is an ι-MPDSE for every game that is plausible for the poisoned dataset.

To formally define the set of plausible Markov Games for a given dataset $\mathcal{D}$, we first need a few definitions.

Definition 2. (Confidence Game Set) The confidence set on the transition function $P_h(s, \boldsymbol{a})$ has the form:

$$\mathrm{CI}_h^P(s, \boldsymbol{a}) := \big\{P_h(s, \boldsymbol{a}) \in \Delta(\mathcal{A}): \\ \|P_h(s, \boldsymbol{a}) - \hat{P}_h(s, \boldsymbol{a})\|_1 \leq \rho_h^P(s, \boldsymbol{a})\big\}$$

where

$$\hat{P}_h(s'|s, \boldsymbol{a}) := \frac{1}{N_h(s, \boldsymbol{a})} \sum_{k=1}^{K} \mathbf{1}_{\{s_{h+1}^{(k)}=s', s_h^{(k)}=s, \boldsymbol{a}_h^{(k)}=\boldsymbol{a}\}}$$

is the maximum likelihood estimate (MLE) of the true transition probability. Similarly, the confidence set on the reward function $R_{i,h}(s, \boldsymbol{a})$ has the form:

$$\mathrm{CI}_{i,h}^R(s, \boldsymbol{a}) := \big\{R_{i,h}(s, \boldsymbol{a}) \in [-b, b]: \\ |R_{i,h}(s, \boldsymbol{a}) - \hat{R}_{i,h}(s, \boldsymbol{a})| \leq \rho_h^R(s, \boldsymbol{a})\big\},$$

where

$$\hat{R}_{i,h}(s, \boldsymbol{a}) := \frac{1}{N_h(s, \boldsymbol{a})} \sum_{k=1}^{K} r_{i,h}^{0,(k)} \mathbf{1}_{\{s_h^{(k)}=s, \boldsymbol{a}_h^{(k)}=\boldsymbol{a}\}}$$ is the MLE of the reward. Then, the set of all plausible Markov Games consistent with $\mathcal{D}$, denoted by CI^G, is defined to be:

$$\mathrm{CI}^G := \big\{G = (\mathcal{S}, \mathcal{A}, P, R, H, \mu): P_h(s, \boldsymbol{a}) \in \mathrm{CI}_h^P(s, \boldsymbol{a}), \\ R_{i,h}(s, \boldsymbol{a}) \in \mathrm{CI}_{i,h}^R(s, \boldsymbol{a}), \forall\, i, h, s, \boldsymbol{a}\big\}.$$

Note that both the attacker and the learners know that all of the rewards are bounded within $[-b, b]$ (we allow $b = \infty$). The values of $\rho_h^P(s, \boldsymbol{a})$ and $\rho_h^R(s, \boldsymbol{a})$ are typically given by concentration inequalities. One standard choice takes the Hoeffding-type form $\rho_h^P(s, \boldsymbol{a}) \propto 1/\sqrt{\max\{N_h(s, a), 1\}}$, and $\rho_h^R(s, \boldsymbol{a}) \propto 1/\sqrt{\max\{N_h(s, a), 1\}}$, where we recall that $N_h(s, a)$ is the visitation count of the state-action pair (s, a) (Xie et al. 2020; Cui and Du 2022; Zhong et al. 2022). We remark that with proper choice of ρ_h^P and ρ_h^R, CI^G contains the game constructed by optimistic MARL algorithms with upper confidence bounds (Xie et al. 2020), as well as that by pessimistic algorithms with lower confidence bounds (Cui and Du 2022; Zhong et al. 2022). See the appendix for details.

With the above definition, we consider an attacker that attempts to modify the original dataset $\mathcal{D}$ into $\mathcal{D}^\dagger$ so that $\boldsymbol{\pi}^\dagger$ is an ι-MPDSE for every plausible game in CI^G induced by the poisoned $\mathcal{D}^\dagger$. This would guarantee the learners adopt $\boldsymbol{\pi}^\dagger$.

The full coverage Assumption 1 is necessary for the above attack goal, as shown in the following proposition. We defer the proof to the appendix.

Proposition 1. *If $N_h(s, \boldsymbol{a}) = 0$ for some $(h, s, \boldsymbol{a})$, then there exist MARL learners for which the attacker's problem is infeasible.*

Poisoning Framework

In this section, we first argue that naively applying single-agent poisoning attacks separately to each agent results in suboptimal attack cost. We then present a new optimal poisoning framework that accounts for multiple agents and thereby allows for efficiently solving the attack problem.

$\mathcal{A}_1 \setminus \mathcal{A}_2$	1	2
1	$(3,3)$	$(1,2)$
2	$(2,1)$	$(0,0)$

Table 1: Single-agent attack reduction example

$\mathcal{A}_i$	r
1	$\{3,1\}$
2	$\{2,0\}$

Table 2: Single-agent attack reduction

Suboptimality of single-agent attack reduction. As a first attempt, the attacker could try to use existing single-agent RL reward poisoning methods. However, this approach is doomed to be suboptimal. Consider the game in Table 1 with $n = 2$ learners, one period, and one state.

Suppose that the original dataset $\mathcal{D}$ has full coverage. For simplicity, we assume that each $(s, \boldsymbol{a})$ pair appears sufficiently many times so that ρ^R is small. In this case, the target policy $\boldsymbol{\pi}^\dagger = (1,1)$ is already an MPDSE, so no reward modification is needed. However, if we use a single-agent approach, each learner i will observe the dataset in Table 2. In this case, to learner i it is not immediately clear which of the two actions is strictly better, for example, when $1, 2$ appears relatively more often than $3, 0$. To ensure that both players take action 1, the attacker needs to modify at least one of the rewards for each player, thus incurring a nonzero (and thus suboptimal) attack cost.

The example above shows that a new approach is needed to construct an optimal poisoning framework tailored to the multi-agent setting. Below we develop such a framework, first for the simple Bandit Game setting, which is then generalized to Markov Games.

Bandit Game Setting

As a stepping stone, we start with a subclass of Markov Games with $|\mathcal{S}| = 1$ and $H = 1$, which are sometimes called bandit games. A bandit game consists of a single-stage normal-form game. For now, we also pretend that the learners simply use the data to compute an MLE point estimate $\hat{G}$ of the game and then solve the estimated game $\hat{G}$. This is unrealistic, but it highlights the attacker's strategy to enforce that $\boldsymbol{\pi}^\dagger$ is an ι-strict DSE in $\hat{G}$.

Suppose the original dataset is $\mathcal{D} = \left\{(\boldsymbol{a}^{(k)}, \boldsymbol{r}^{0,(k)})\right\}_{k=1}^K$ (recall we no longer have state or period). Also, let $N(\boldsymbol{a}) := \sum_{k=1}^K \mathbf{1}_{\{\boldsymbol{a}^{(k)}=\boldsymbol{a}\}}$ be the action counts. The attacker's problem can be formulated as a convex optimization problem given in (1).

$$
\begin{aligned}
&\min_{r^\dagger} C\big(r^0, r^\dagger\big) \\
&\text{s.t. } R^\dagger(\boldsymbol{a}) := \frac{1}{N(\boldsymbol{a})} \sum_{k=1}^K \boldsymbol{r}^{\dagger,(k)} \mathbf{1}_{\left\{\boldsymbol{a}^{(k)}=\boldsymbol{a}\right\}}, \forall\, \boldsymbol{a}; \\
&\quad R_i^\dagger\big(\pi_i^\dagger, a_{-i}\big) \geq R_i^\dagger\left(a_i, a_{-i}\right) + \iota, \forall\, i, a_{-i}, a_i \neq \pi_i^\dagger; \\
&\quad \boldsymbol{r}^{\dagger,(k)} \in [-b, b]^n, \forall\, k.
\end{aligned} \tag{1}
$$

The first constraint in (1) models the learners' MLE $\hat{G}$ after poisoning. The second constraint enforces that $\boldsymbol{\pi}^\dagger$ is an ι-strict DSE of $\hat{G}$ by definition. We observe that:

1. The problem is feasible if $\iota \leq 2b$, since the attacker can always set, for each agent, the reward to be b for the target action and $-b$ for all other actions;
2. If the cost function $C(\cdot, \cdot)$ is the L^1-norm, the problem is a linear program (LP) with nK variables and $(A-1)A^{n-1} + 2nK$ inequality constraints (assuming each learner has $|\mathcal{A}_i| = A$ actions);
3. After the attack, learner i only needs to see its own rewards to be convinced that $\pi_i^\dagger$ is a dominant strategy; learner i does not need to observe other learners' rewards.

This simple formulation serves as an asymptotic approximation to the attack problem for confidence-bound-based learners. In particular, when $N(\boldsymbol{a})$ is large for all $\boldsymbol{a}$, the confidence intervals on P and R are usually small.

With the above idea in place, we can consider more realistic learners that are uncertainty-aware. For these learners, the attacker attempts to enforce an ι separation between the lower bound of the target action's reward and the upper bounds of all other actions' rewards (similar to arm elimination in bandits). With such separation, all plausible games in CI^G would have the target action profile as the dominant strategy equilibrium. This approach can be formulated as a slightly more complex optimization problem (2), where the second and third constraints enforce the desired ι separation. The formulation (2) can be solved using standard optimization solvers, hence the optimal attack can be computed efficiently.

$$
\begin{aligned}
&\min_{r^\dagger} C(r^0, r^\dagger) \\
&\text{s.t. } R^\dagger(\boldsymbol{a}) := \frac{1}{N(a)} \sum_{k=1}^K \boldsymbol{r}^{\dagger,(k)} \mathbf{1}_{\left\{\boldsymbol{a}^{(k)}=a\right\}}, \forall\, \boldsymbol{a}; \\
&\quad \text{CI}_i^{R^\dagger}(\boldsymbol{a}) := \big\{R_i(\boldsymbol{a}) \in [-b, b] : \big|R_i(\boldsymbol{a}) - R_i^\dagger(\boldsymbol{a})\big| \\
&\qquad\qquad \leq \rho^R(\boldsymbol{a})\big\}, \quad \forall\, i, \boldsymbol{a}; \\
&\quad \min_{R_i \in \text{CI}_i^{R^\dagger}(\pi_i^\dagger, a_{-i})} R_i \geq \max_{R_i \in \text{CI}_i^{R^\dagger}(a_i, a_{-i})} R_i + \iota, \\
&\qquad\qquad \forall\, i, a_{-i}, a_i \neq \pi_i^\dagger; \\
&\quad \boldsymbol{r}^{\dagger,(k)} \in [-b, b]^n, \forall\, k.
\end{aligned} \tag{2}
$$

We next consider whether this formulation has a feasible solution. Below we characterize the feasibility of the at-

tack in terms of the margin parameter ι and the confidence bounds.

Proposition 2. *The attacker's problem* (2) *is feasible if* $\iota \leq 2b - 2\rho^R(\boldsymbol{a}), \forall\, \boldsymbol{a} \in \mathcal{A}$.

Proposition 2 is a special case of the general Theorem 5 with $H = |\mathcal{S}| = 1$. We note that the condition in Proposition 2 has an equivalent form that relates to the structure of the dataset. We later present this form for a more general case.

When an L^1-norm cost function is used, we show in the appendix that the formulation (2) can also be efficiently solved.

Proposition 3. *With* L^1*-norm cost function* $C(\cdot,\cdot)$*, the problem* (2) *can be formulated as a linear program.*

Markov Game Setting

We now generalize the ideas from the bandit setting to derive a poisoning framework for arbitrary Markov Games. With multiple states and periods, there are two main complications:

1. In each period h, the learners' decision depends on Q_h, which involves both the immediate reward R_h and the future return Q_{h+1};
2. The uncertainty in Q_h amplifies as it propagates backward in h.

Accordingly, the attacker needs to design the poisoning attack recursively.

Our main technical innovation is an attack formulation based on Q *confidence-bound backward induction*. The attacker maintains confidence upper and lower bounds on the learners' Q function, $\overline{Q}$, and $\underline{Q}$, with backward induction. To ensure $\boldsymbol{\pi}^\dagger$ becomes an ι-MPDSE, the attacker again attempts to ι-separate the lower bound of the target action and the upper bound of all other actions, at all states and periods.

Recall Definition 2: given the training dataset $\mathcal{D}$, one can compute the MLEs $\boldsymbol{R}_h$ and corresponding confidence sets $\mathrm{CI}^R_{i,h}$ for the reward. The attacker aims to poison $\mathcal{D}$ into $\mathcal{D}^\dagger$ so that the MLEs and confidence sets become $\boldsymbol{R}^\dagger_h$ and $\mathrm{CI}^{R^\dagger}_{i,h}$, under which $\boldsymbol{\pi}^\dagger$ is the unique ι-MPDSE for all plausible games in the corresponding confidence game set. The attacker finds the minimum cost way of doing so by solving a Q confidence-bound backward induction optimization problem, given in (3)–(7).

$$\min_{r^\dagger} C\left(r^0, r^\dagger\right) \tag{3}$$

$$\text{s.t. } R^\dagger_{i,h}(s,\boldsymbol{a}) := \frac{1}{N_h(s,\boldsymbol{a})}\sum_{k=1}^{K} r^{\dagger,(k)}_{i,h}\mathbf{1}_{\left\{s_h^{(k)}=s,\boldsymbol{a}_h^{(k)}=\boldsymbol{a}\right\}},$$
$$\forall\, h,s,i,\boldsymbol{a}$$
$$\mathrm{CI}^{R^\dagger}_{i,h}(s,\boldsymbol{a}) := \Big\{R_{i,h}(s,\boldsymbol{a}) \in [-b,b]$$
$$: \left|R_{i,h}(s,\boldsymbol{a}) - R^\dagger_{i,h}(s,\boldsymbol{a})\right| \leq \rho^R_h(s,\boldsymbol{a})\Big\},$$
$$\forall\, h,s,i,\boldsymbol{a}$$

$$\underline{Q}_{i,H}(s,\boldsymbol{a}) := \min_{R_{i,H}\in \mathrm{CI}^{R^\dagger}_{i,H}(s,\boldsymbol{a})} R_{i,H}, \forall\, s,i,\boldsymbol{a}$$
$$\underline{Q}_{i,h}(s,\boldsymbol{a}) := \min_{R_{i,h}\in \mathrm{CI}^{R^\dagger}_{i,h}(s,\boldsymbol{a})} R_{i,h}$$
$$+ \min_{P_h\in \mathrm{CI}^{P}_{h}(s,\boldsymbol{a})} \sum_{s'\in\mathcal{S}} P_h(s')\,\underline{Q}_{i,h+1}\left(s', \boldsymbol{\pi}^\dagger_{h+1}(s')\right),$$
$$\forall\, h < H, s,i,\boldsymbol{a} \tag{4}$$

$$\overline{Q}_{i,H}(s,\boldsymbol{a}) := \max_{R_{i,H}\in \mathrm{CI}^{R^\dagger}_{i,H}(s,\boldsymbol{a})} R_{i,H}, \forall\, s,i,\boldsymbol{a}$$
$$\overline{Q}_{i,h}(s,\boldsymbol{a}) := \max_{R_{i,h}\in \mathrm{CI}^{R^\dagger}_{i,h}(s,\boldsymbol{a})} R_{i,h}$$
$$+ \max_{P_h\in \mathrm{CI}^{P}_{h}(s,\boldsymbol{a})} \sum_{s'\in\mathcal{S}} P_h(s')\,\overline{Q}_{i,h+1}\left(s', \boldsymbol{\pi}^\dagger_{h+1}(s')\right),$$
$$\forall\, h < H, s,i,\boldsymbol{a} \tag{5}$$

$$\underline{Q}_{i,h}\left(s, \left(\pi^\dagger_{i,h}(s), a_{-i}\right)\right) \geq \overline{Q}_{i,h}(s,(a_i,a_{-i})) + \iota,$$
$$\forall\, h,s,i,a_{-i},a_i \neq \pi^\dagger_{i,h}(s) \tag{6}$$

$$\boldsymbol{r}^{\dagger,(k)}_h \in [-b,b]^n, \forall\, h,k. \tag{7}$$

The backward induction steps (4) and (5) ensure that $\underline{Q}$ and $\overline{Q}$ are valid lower and upper bounds for the Q function for all plausible Markov Games in CI^G, for all periods. The margin constraints (6) enforce an ι-separation between the target action and other actions at all states and periods. We emphasize that the agents need not consider Q at all in their learning algorithm; Q only appears in the optimization due to its presence in the definition of MPDSE.

Again, pairing an efficient optimization solver with the above formulation gives an efficient algorithm for constructing the poisoning. We now answer the important questions of whether this formulation admits a feasible solution and whether these solutions yield successful attacks. The lemma below provides a positive answer to the second question.

Lemma 4. *If the attack formulation* (3)–(7) *is feasible,* $\boldsymbol{\pi}^\dagger$ *is the unique* ι*-MPDSE of every Markov Game* $G \in \mathrm{CI}^G$.

Moreover, the attack formulation admits feasible solutions under mild conditions on the dataset.

Theorem 5. *The attacker formulation* (3)–(7) *is feasible if the following condition holds:*

$$\iota \leq 2b - (H+1)\,\rho^R_h(s,\boldsymbol{a}), \quad \forall\, h \in [H], s \in \mathcal{S}, \boldsymbol{a} \in \mathcal{A}.$$

We remark that the learners know the upper bound b and may use it to exclude implausible games. The accumulation of confidence intervals over the H periods results in the extra factor $(H+1)$ on ρ^R_h. Theorem 5 implies that the problem is feasible so long as the dataset is sufficiently populated; that is, each (s,a) pair should appear frequently enough to have a small confidence interval half-width ρ^R_h. The following corollary provides a precise condition on the visit accounts that guarantees feasibility.

Corollary 6. *Given a confidence probability* δ *and the confidence interval half-width* $\rho^R_h(s,\boldsymbol{a}) = f(\frac{1}{N_h(s,a)})$ *for some*

strictly increasing function f, the condition in Theorem 5 holds if

$$N_h(s,\boldsymbol{a}) \geq \left(f^{-1}\left(\frac{2b-\iota}{H+1}\right)\right)^{-1}.$$

In particular, for the natural choice of Hoeffding-type $\rho_h^R(s,\boldsymbol{a}) = 2b\sqrt{\frac{\log\left(\left(H\left|\mathcal{S}\right|\left|\mathcal{A}\right|\right)/\delta\right)}{\max\left\{N_h\left(s,\boldsymbol{a}\right),1\right\}}}$*, it suffices that,*

$$N_h(s,\boldsymbol{a}) \geq \frac{4b^2\left(H+1\right)^2\log\left(\left(H\left|\mathcal{S}\right|\left|\mathcal{A}\right|\right)/\delta\right)}{\left(2b-\iota\right)^2}.$$

Despite the inner min and max in the problem (3)–(7), the problem can be formulated as an LP, thanks to LP duality.

Theorem 7. *With L^1-norm cost function $C(\cdot,\cdot)$, problem* (3)–(7) *can be formulated as an LP.*

The proofs of the above results can be found in the appendix.

Cost Analysis

Now that we know how the attacker can poison the dataset in the multi-agent setting, we can study the structure of attacks. The structure is most easily seen by analyzing the minimal attack cost. To this end, we give general bounds that relate the minimal attack cost to the structure of the underlying Markov Game. The attack cost upper bounds show which games are particularly susceptible to poison, and the attack cost lower bounds demonstrate that some games are expensive to poison.

Overview of results: Specifically, we shall present two types of upper/lower bounds on the attack cost: (i) *universal bounds* that hold for all attack problem instances simultaneously; (ii) *instance-dependent bounds* that are stated in terms of certain properties of the instance. We also discuss problem instances under which these two types of bounds are tight and coincide with each other.

We note that all bounds presented here are with respect to the L^1-cost, but many of them generalize to other cost functions, especially the L^∞-cost. The proofs of the results presented in this section are provided in the appendix.

Setup: Let $I = (\mathcal{D}, \boldsymbol{\pi}^\dagger, \rho^R, \rho^P, \iota)$ denote an instance of the attack problem, and $\hat{G}$ denote the corresponding MLE of the Markov Game derived from $\mathcal{D}$. We denote by $I_h = (\mathcal{D}_h, \boldsymbol{\pi}_h^\dagger, \rho_h^R, \rho_h^P, \iota)$ the restriction of the instance to period h. In particular, $\hat{R}_h(s)$ derived from $\mathcal{D}_h$ is exactly the normal-form game at state s and period h of $\hat{G}$. We define $C^*(I)$ to be the optimal L^1-poisoning cost for the instance I; that is, $C^*(I)$ is the optimal value of the optimization problem (3)–(7) evaluated on I. We say the attack instance I is *feasible* if this optimization problem is feasible. If I is infeasible, we define $C^*(I) = \infty$. WLOG, we assume that $|\mathcal{A}_1| = \cdots = |\mathcal{A}_n| = A$. In addition, we define the minimum visit count for each period h in $\mathcal{D}$ as $\underline{N}_h := \min_{s\in\mathcal{S}}\min_{\boldsymbol{a}\in\mathcal{A}} N_h\left(s,\boldsymbol{a}\right)$, and the minimum over all periods as $\underline{N} := \min_{h\in H}\underline{N}_h$. We similarly define the maximum visit counts as $\overline{N}_h = \max_{s\in\mathcal{S}}\max_{\boldsymbol{a}\in\mathcal{A}} N_h\left(s,\boldsymbol{a}\right)$ and $\overline{N} = \max_h \overline{N}_h$. Lastly, we define $\underline{\rho} = \min_{h,s,\boldsymbol{a}} \rho_h^R(s,\boldsymbol{a})$ and $\overline{\rho} = \max_{h,s,\boldsymbol{a}} \rho_h^R(s,\boldsymbol{a})$, the minimum and maximum confidence half-width.

$\mathcal{A}_1/\mathcal{A}_2$	1	2	...	$\lvert\mathcal{A}_2\rvert$
1	$-b,-b$	$-b,b$	...	$-b,b$
2	$b,-b$	b,b	...	b,b
...	...	...	...	...
$\lvert\mathcal{A}_1\rvert$	$b,-b$	b,b	...	b,b

Table 3: MLE $\hat{\boldsymbol{R}}_h(s,\cdot)$ before attack

$\mathcal{A}_1/\mathcal{A}_2$	1	...	$2,...,\lvert\mathcal{A}_2\rvert$
1	b,b	...	$b,b-2\rho-\iota$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$2,...,\lvert\mathcal{A}_1\rvert$	$b-2\rho-\iota,b$	...	$b-2\rho-\iota,b-2\rho-\iota$

Table 4: MLE $\hat{\boldsymbol{R}}_h(s,\cdot)$ after attack

Universal Cost Bounds

With the above definitions, we present universal attack cost bounds that hold simultaneously for all attack instances.

Theorem 8. *For any feasible attack instance I, we have that,*

$$0 \leq C^*(I) \leq \overline{N}H|\mathcal{S}|nA^n 2b.$$

As these upper and lower bounds hold for all instances, they are typically loose. However, they are nearly tight. If $\boldsymbol{\pi}^\dagger$ is already an ι-MPDSE for all plausible games, then no change to the rewards is needed and the attack cost is 0, hence the lower bound is tight for such instances. We can also construct a high-cost instance to show the near-tightness of the upper bound.

Specifically, consider the dataset for a bandit game, $\mathcal{D} = \left\{(\boldsymbol{a}^{(k)}, \boldsymbol{r}^{0,(k)})\right\}_{k=1}^K$, where $\mathcal{A} = A^n$ and each action appears exactly N times, i.e., $\overline{N} = \underline{N} = N$ and $K = NA^n$. The target policy is $\boldsymbol{\pi}^\dagger = (1,\ldots,1)$. The dataset is constructed so that $\boldsymbol{r}_i^{0,(k)} = -b$ if $\boldsymbol{a}_i^{(k)} = \pi_{i,h}^\dagger(s)$ and $\boldsymbol{r}_i^{0,(k)} = b$ otherwise. These rewards are essentially the extreme opposite of what the attacker needs to ensure $\boldsymbol{\pi}^\dagger$ is an ι-DSE. Note, the dataset induces the MLE of the game shown in Table 3 for the special case with $n = 2$ players.

For simplicity, suppose that the same confidence half-width $\rho^R(\boldsymbol{a}) = \rho < b$ is used for all $\boldsymbol{a}$. Let $\iota \in (0,b)$ be arbitrary. For this instance, to install $\boldsymbol{\pi}^\dagger$ as the ι-DSE, the attacker can flip all rewards in a way that is illustrated in Table 4, inducing a cost as the upper bound in Theorem 8. The situation is the same for $n \geq 2$ learners. Our instance-dependent lower bound, presented later in Theorem 12, implies that any attack on this instance must have cost at least $NnA^{n-1}(2b+2\rho+\iota)$. This lower bound matches the refined upper bound in the proof of Theorem 9, implying the refined bounds are tight for this instance. Noticing that the universal bound in Theorem 8 only differs by an $O(A)$-factor implies it is nearly tight.

Instance-Dependent Cost Bounds

Next, we derive general bounds on the attack cost that depends on the structure of the underlying instance. Our strategy is to reduce the problem of bounding Markov Game

costs to the easier problem of bounding Bandit Game costs. We begin by showing that the cost of poisoning a Markov Game dataset can be bounded in terms of the cost of poisoning the datasets corresponding to its individual period games.

Theorem 9. *For any feasible attack instance I, we have that $C^*(I_H) \leq C^*(I)$ and,*

$$C^*(I) \leq \sum_{h=1}^{H} C^*(I_h) + 2bnH|\mathcal{S}|\overline{N} + H^2\overline{\rho}|\mathcal{S}|nA^n\overline{N}$$

Here we see the effect of the learner's uncertainty. If ρ^R is small, then poisoning costs slightly more than poisoning each bandit instance independently. This is desirable since it allows the attacker to solve the much easier bandit instances instead of the full problem.

The lower bound is valid for all Markov Games, but it is weak in that it only uses the last period cost. However, this is the most general lower bound one can obtain without additional assumptions on the structure of the game. If we assume additional structure on the dataset, then the above lower bound can be extended beyond the last period, forcing a higher attack cost.

Lemma 10. *Let I be any feasible attack instance containing at least one uniform transition in CI_h^P for each period h, i.e., there is some $\hat{P}_h(s' \mid s, \boldsymbol{a}) \in \text{CI}_h^P$ with $\hat{P}_h(s' \mid s, \boldsymbol{a}) = 1/|\mathcal{S}|, \forall h, s', s, \boldsymbol{a}$. Then, we have that*

$$C^*(I) \geq \sum_{h=1}^{H} C^*(I_h).$$

In words, for these instances the optimal cost for poisoning is not too far off from the optimal cost of poisoning each period game independently. We note this is where the effects of ρ^P show themselves. If the dataset is highly uncertain on the transitions, it becomes likely that a uniform transition exists in CI^P. Thus, a higher ρ^P leads to a higher cost and effectively devolves the set of plausible games into a series of independent games.

Now that we have the above relationships, we can focus on bounding the attack cost for bandit games. To be precise, we bound the cost of poisoning a period game instance I_h. To this end, we define ι-dominance gaps.

Definition 3. (Dominance Gaps) For every $h \in [H], s \in \mathcal{S}, i \in [n]$ and $a_{-i} \in \mathcal{A}_{-i}$, the ι-dominance gap, $d_{i,h}^{\iota}(s, a_{-i})$, is defined as

$$d_{i,h}^{\iota}(s, a_{-i}) :=$$
$$\Big[\max_{a_i \neq \pi_{i,h}^{\dagger}(s)} \Big[\hat{R}_{i,h}\big(s,(a_i,a_{-i})\big) + \rho_h^R\big(s,(a_i,a_{-i})\big)\Big]$$
$$- \hat{R}_{i,h}\Big(s,\big(\pi_{i,h}^{\dagger}(s),a_{-i}\big)\Big) + \rho_h^R\Big(s,\big(\pi_{i,h}^{\dagger}(s),a_{-i}\big)\Big) + \iota\Big]_+$$

where $\hat{R}$ is the MLE w.r.t. the original dataset $\mathcal{D}$.

The dominance gaps measure the minimum amount by which the attacker would have to increase the reward for learner i while others are playing a_{-i}, so that the action $\pi_{i,h}^{\dagger}(s)$ becomes ι-dominant for learner i. We then consolidate all the dominance gaps for period h into the variable $\Delta_h(\iota)$,

$$\Delta_h(\iota) := \sum_{s\in S}\sum_{i=1}^{n}\sum_{a_{-i}}\Big(d_{i,h}^{\iota}(s,a_{-i}) + \delta_{i,h}^{\iota}(s,a_{-i})\Big)$$

Where $\delta_{i,h}^{\iota}(s,a_{-i})$ is a minor overflow term defined in the appendix. With all this machinery set up, we can give precise bounds on the minimal cost needed to attack a single-period game.

Lemma 11. *The optimal attack cost for I_h satisfies*

$$\underline{N}_h\Delta_h(\iota) \leq C^*(I_h) \leq \overline{N}_h\Delta_h(\iota).$$

Combining these bounds with Theorem 9 gives complete attack cost bounds for general Markov game instances.

The lower bounds in both Lemma 10 and Lemma 11 expose an exponential dependency on n, the number of players, for some datasets $\mathcal{D}$. These instances essentially require the attacker to modify $\hat{R}_{i,h}(s, \boldsymbol{a})$ for every $\boldsymbol{a} \in \mathcal{A}$. A concrete instance can be constructed by taking the high-cost dataset derived as the tight example before and extending it into a general Markov Game. We simply do this by giving the game several identical states and uniform transitions. In terms of the dataset, each episode consists of independent plays of the same normal-form game, possibly with a different state observed. For this dataset the ι-dominance gap can be shown to be $d_{i,h}^{\iota}(s,a_{-i}) = 2b + 2\rho + \iota$. A direct application of Lemma 10 gives the following explicit lower bound.

Theorem 12. *There exists a feasible attack instance I for which it holds that*

$$C^*(I) \geq \underline{N}H\,|\mathcal{S}|\,nA^{n-1}\,(2b + 2\rho + \iota)\,.$$

Recall the attacker wants to assume little about the learners and therefore chooses to install an ι-MPDSE (instead of making stronger assumptions on the learners and installing a Nash equilibrium or a non-Markov perfect equilibrium). On some datasets $\mathcal{D}$, the exponential poisoning cost is the price the attacker pays for this flexibility.

Conclusion

We studied a security threat to offline MARL where an attacker can force learners into executing an arbitrary Dominant Strategy Equilibrium by minimally poisoning historical data. We showed that the attack problem can be formulated as a linear program, and provided an analysis on the attack feasibility and cost. This paper thus helps to raise awareness of the trustworthiness of multi-agent learning. We encourage the community to study defense against such attacks, e.g. via robust statistics and reinforcement learning.

Acknowledgements

McMahan is supported in part by NSF grant 2023239. Zhu is supported in part by NSF grants 1545481, 1704117, 1836978, 2023239, 2041428, 2202457, ARO MURI W911NF2110317, and AF CoE FA9550-18-1-0166. Xie is partially supported by NSF grant 1955997 and JP Morgan Faculty Research Awards. We also thank Yudong Chen for his useful comments and discussions.

References

Anderson, A.; Shoham, Y.; and Altman, A. 2010. Internal implementation. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*, 191–198. Citeseer.

Banihashem, K.; Singla, A.; Gan, J.; and Radanovic, G. 2022. Admissible Policy Teaching through Reward Design. *arXiv preprint arXiv:2201.02185*.

Banihashem, K.; Singla, A.; and Radanovic, G. 2021. Defense against reward poisoning attacks in reinforcement learning. *arXiv preprint arXiv:2102.05776*.

Bogunovic, I.; Losalka, A.; Krause, A.; and Scarlett, J. 2021. Stochastic linear bandits robust to adversarial attacks. In *International Conference on Artificial Intelligence and Statistics*, 991–999. PMLR.

Cui, Q.; and Du, S. S. 2022. When is Offline Two-Player Zero-Sum Markov Game Solvable? *arXiv preprint arXiv:2201.03522*.

Garcelon, E.; Roziere, B.; Meunier, L.; Teytaud, O.; Lazaric, A.; and Pirotta, M. 2020. Adversarial Attacks on Linear Contextual Bandits. *arXiv preprint arXiv:2002.03839*.

Gleave, A.; Dennis, M.; Wild, C.; Kant, N.; Levine, S.; and Russell, S. 2019. Adversarial policies: Attacking deep reinforcement learning. *arXiv preprint arXiv:1905.10615*.

Guan, Z.; Ji, K.; Bucci Jr, D. J.; Hu, T. Y.; Palombo, J.; Liston, M.; and Liang, Y. 2020. Robust stochastic bandit algorithms under probabilistic unbounded adversarial attack. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 4036–4043.

Guo, W.; Wu, X.; Huang, S.; and Xing, X. 2021. Adversarial policy learning in two-player competitive games. In *International Conference on Machine Learning*, 3910–3919. PMLR.

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

Jiang, J.; and Lu, Z. 2021. Offline decentralized multi-agent reinforcement learning. *arXiv preprint arXiv:2108.01832*.

Jun, K.-S.; Li, L.; Ma, Y.; and Zhu, J. 2018. Adversarial attacks on stochastic bandits. *Advances in Neural Information Processing Systems*, 31: 3640–3649.

Littman, M. L. 1994. Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994*, 157–163. Elsevier.

Liu, F.; and Shroff, N. 2019. Data poisoning attacks on stochastic bandits. In *International Conference on Machine Learning*, 4042–4050. PMLR.

Liu, G.; and Lai, L. 2021. Provably Efficient Black-Box Action Poisoning Attacks Against Reinforcement Learning. *Advances in Neural Information Processing Systems*, 34.

Lu, S.; Wang, G.; and Zhang, L. 2021. Stochastic Graphical Bandits with Adversarial Corruptions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 8749–8757.

Lykouris, T.; Simchowitz, M.; Slivkins, A.; and Sun, W. 2021. Corruption-robust exploration in episodic reinforcement learning. In *Conference on Learning Theory*, 3242–3245. PMLR.

Ma, Y.; Jun, K.-S.; Li, L.; and Zhu, X. 2018. Data poisoning attacks in contextual bandits. In *International Conference on Decision and Game Theory for Security*, 186–204. Springer.

Ma, Y.; Wu, Y.; and Zhu, X. 2021. Game Redesign in No-regret Game Playing. *arXiv preprint arXiv:2110.11763*.

Ma, Y.; Zhang, X.; Sun, W.; and Zhu, J. 2019. Policy poisoning in batch reinforcement learning and control. *Advances in Neural Information Processing Systems*, 32: 14570–14580.

Monderer, D.; and Tennenholtz, M. 2004. k-Implementation. *Journal of Artificial Intelligence Research*, 21: 37–62.

Pan, L.; Huang, L.; Ma, T.; and Xu, H. 2022. Plan better amid conservatism: Offline multi-agent reinforcement learning with actor rectification. In *International Conference on Machine Learning*, 17221–17237. PMLR.

Rakhsha, A.; Radanovic, G.; Devidze, R.; Zhu, X.; and Singla, A. 2020. Policy teaching via environment poisoning: Training-time adversarial attacks against reinforcement learning. In *International Conference on Machine Learning*, 7974–7984. PMLR.

Rakhsha, A.; Radanovic, G.; Devidze, R.; Zhu, X.; and Singla, A. 2021a. Policy teaching in reinforcement learning via environment poisoning attacks. *Journal of Machine Learning Research*, 22(210): 1–45.

Rakhsha, A.; Zhang, X.; Zhu, X.; and Singla, A. 2021b. Reward poisoning in reinforcement learning: Attacks against unknown learners in unknown environments. *arXiv preprint arXiv:2102.08492*.

Rangi, A.; Tran-Thanh, L.; Xu, H.; and Franceschetti, M. 2022a. Saving stochastic bandits from poisoning attacks via limited data verification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 8054–8061.

Rangi, A.; Xu, H.; Tran-Thanh, L.; and Franceschetti, M. 2022b. Understanding the Limits of Poisoning Attacks in Episodic Reinforcement Learning. In Raedt, L. D., ed., *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 3394–3400. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Shapley, L. S. 1953. Stochastic games. *Proceedings of the national academy of sciences*, 39(10): 1095–1100.

Sun, Y.; Huo, D.; and Huang, F. 2020. Vulnerability-aware poisoning mechanism for online rl with unknown dynamics. *arXiv preprint arXiv:2009.00774*.

Wei, C.-Y.; Dann, C.; and Zimmert, J. 2022. A model selection approach for corruption robust reinforcement learning. In *International Conference on Algorithmic Learning Theory*, 1043–1096. PMLR.

Wu, F.; Li, L.; Xu, C.; Zhang, H.; Kailkhura, B.; Kenthapadi, K.; Zhao, D.; and Li, B. 2022. COPA: Certifying Robust Policies for Offline Reinforcement Learning against Poisoning Attacks. *arXiv preprint arXiv:2203.08398*.

Xie, Q.; Chen, Y.; Wang, Z.; and Yang, Z. 2020. Learning zero-sum simultaneous-move markov games using function approximation and correlated equilibrium. In *Conference on learning theory*, 3674–3682. PMLR.

Yang, L.; Hajiesmaili, M.; Talebi, M. S.; Lui, J.; and Wong, W. S. 2021. Adversarial Bandits with Corruptions: Regret Lower Bound and No-regret Algorithm. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Zhang, H.; and Parkes, D. C. 2008. Value-Based Policy Teaching with Active Indirect Elicitation. In *AAAI*, volume 8, 208–214.

Zhang, H.; Parkes, D. C.; and Chen, Y. 2009. Policy teaching through reward function learning. In *Proceedings of the 10th ACM conference on Electronic commerce*, 295–304.

Zhang, K.; Yang, Z.; and Başar, T. 2021. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of Reinforcement Learning and Control*, 321–384.

Zhang, X.; Chen, Y.; Zhu, J.; and Sun, W. 2021a. Corruption-robust offline reinforcement learning. *arXiv preprint arXiv:2106.06630*.

Zhang, X.; Chen, Y.; Zhu, X.; and Sun, W. 2021b. Robust policy gradient against strong data corruption. In *International Conference on Machine Learning*, 12391–12401. PMLR.

Zhang, X.; Ma, Y.; Singla, A.; and Zhu, X. 2020. Adaptive reward-poisoning attacks against reinforcement learning. In *International Conference on Machine Learning*, 11225–11234. PMLR.

Zhong, H.; Xiong, W.; Tan, J.; Wang, L.; Zhang, T.; Wang, Z.; and Yang, Z. 2022. Pessimistic minimax value iteration: Provably efficient equilibrium learning from offline datasets. *arXiv preprint arXiv:2202.07511*.

Zuo, S. 2020. Near Optimal Adversarial Attack on UCB Bandits. *arXiv preprint arXiv:2008.09312*.