CAN REINFORCEMENT LEARNING EFFICIENTLY FIND STACKELBERG-NASH EQUILIBRIA IN GENERAL-SUM MARKOV GAMES?

Anonymous authors
Paper under double-blind review

ABSTRACT

We study multi-player general-sum Markov games with one of the players designated as the leader and the rest regarded as the followers. In particular, we focus on the class of games where the state transitions are only determined by the leader's action while the actions of all the players determine their immediate rewards. For such a game, our goal is to find the Stackelberg-Nash equilibrium (SNE), which is a policy pair (π^*, ν^*) such that (i) π^* is the optimal policy for the leader when the followers always play their best response, and (ii) ν^* is the best response policy of the followers, which is a Nash equilibrium of the followers' game induced by π^*. We develop sample efficient reinforcement learning (RL) algorithms for solving SNE for both the online and offline settings. Respectively, our algorithms are optimistic and pessimistic variants of least-squares value iteration and are readily able to incorporate function approximation for handling large state spaces. Furthermore, for the case with linear function approximation, we prove that our algorithms achieve sublinear regret and suboptimality under online and offline setups respectively. To our best knowledge, we establish the first provably efficient RL algorithms for solving SNE in general-sum Markov games with leader-controlled state transitions.

1 INTRODUCTION

Reinforcement learning (RL) has achieved striking empirical successes in solving complicated real-world sequential decision-making problems (Mnih et al., 2015; Duan et al., 2016; Silver et al., 2016; 2017; 2018; Agostinelli et al., 2019; Akkaya et al., 2019). Motivated by these successes, multi-agent extensions of RL algorithms recently have gained great popularity in decision-making problems involving multiple interacting agents (Busoniu et al., 2008; Hernandez-Leal et al., 2018; 2019; OroojlooyJadid & Hajinezhad, 2019; Zhang et al., 2019). Multi-agent RL is often modeled as a Markov game (Littman, 1994) where, at each time step, each player (agent) takes an action simultaneously at each state of the environment, observe her own immediate reward, and the environment evolves into a next state. Here both the reward of each player and the state transition depends on the actions of all players. From the perspective of each player, her goal is to find a policy that maximizes her expected total reward in the presence of other agents.

In Markov games, depending on the structure of the reward functions, the relationship among the players can be either collaborative, where each player has the same reward function, or competitive, where the sum of the reward function is equal to zero, or mixed, which corresponds to a general-sum game. While most of existing theoretical results focus on the collaborative or two-player competitive settings, the mixed setting is oftentimes more pertinent to real-world multi-agent applications.

Moreover, in addition to having diverse reward functions, the players might also have asymmetric roles in the Markov game — the players might be divided into leaders and followers, where the leaders' joint policy determines a general-sum game for the followers. Games with such a leader-follower structure is popular in applications such as mechanism design (Conitzer & Sandholm, 2002; Roughgarden, 2004; Garg & Narahari, 2005; Kang & Wu, 2014), security games (Tambe, 2011; Korzhyk et al., 2011), incentive design (Zheng et al., 1984; Ratliff et al., 2014; Chen et al., 2016; Ratliff & Fiez, 2020), and model-based RL (Rajeswaran et al., 2020). Consider a simplified economic system that consists of a government and a group of companies, where the companies purchase or sell

goods, and the government collects taxes from transactions. Such a problem can be viewed as a multi-player general-sum game, where the government serves as the leader and the companies are followers (Zheng et al., 2020). In particular, when the government sets a tax rate, the companies form a general-sum game themselves, whose reward functions depend on the tax rate. Each company aims to maximize their own revenue, and thus ideally they achieve a Nash equilibrium (NE) of the induced game. Whereas the goal of the government might be achieving the social welfare, which might be measured via certain fairness metrics computed by the revenues of the companies.

In multi-player Markov games with such a leader-follower structure, the desired solution concept is the Stackelberg-Nash equilibrium (SNE) (Başar & Olsder, 1998). In the setting where there is a single leader, SNE corresponds to a pair of leader's policy π^* and followers' joint policy ν^* that satisfies the following two properties: (i) when the leader adopts π^*, ν^* is the best-response policy of the followers, i.e., ν^* is a Nash equilibrium of the followers' subgame induced by π^*; and (ii) π^* is the optimal policy of the leader assuming the followers always adopt the best response.

We are interested in finding an SNE in a multi-player Markov game when the reward functions and Markov transition kernel are unknown. In particular, we focus on the setting with a single leader and the state transitions only depend on the leader's actions. That is, the followers' actions only affect the rewards received by the leader and followers. For such a game, we are interested in the following question:

Can we develop reinforcement learning methods that provably find Stackelberg-Nash equilibria in leader-controlled general-sum games with sample efficiency?

To this end, we consider both online and offline RL settings, where in the former, we learn the SNE in a trial-an-error fashion by interacting with the environment and generating data, and in the latter, we learn the SNE from a given dataset that is collected a priori. For the online setting, as the transition model is unknown, to achieve sample efficiency, the equilibrium-finding algorithm also needs to take the exploration-exploitation tradeoff into consideration. Although the similar challenge has been studied in zero-sum Markov game, it seems unclear how to incorporate popular exploration mechanisms such as optimism in the face of uncertainty (Sutton & Barto, 2018) into SNE finding. Meanwhile, under the offline setting, as the RL agent has no control of data collection, it is ideal to design an RL algorithm with theoretical guarantees for an arbitrary dataset that might not be sufficiently explorative.

Our contributions Our contributions are three-fold. First, for the episodic leader-controlled general-sum game, under the online and offline settings respectively, we propose optimistic and pessimistic variants of the least-squares value iteration (LSVI) algorithm. In particular, in a version of LSVI, we estimate the optimal action-value function of the leader via least-squares regression and construct an estimate of the SNE by solving the SNE of the multi-matrix game for each state, whose payoff matrices are given by the leader's estimated action-value function and the followers' reward functions. Moreover, we add a UCB exploration bonus to the least-squares solution to achieve optimism in the online setting. Whereas in the offline setting, pessimism is achieved by subtracting a penalty function constructed using the offline data, which is equal to the negative bonus function. Moreover, these algorithms are readily able to incorporate function approximators and we showcase the version with linear function approximation. Second, under the online setting, we prove that our optimistic LSVI algorithm achieves a sublinear $\widetilde{\mathcal{O}}(H^2\sqrt{d^3K})$ regret, where K is the number of episodes, H is the horizon, d is the dimension of the feature mapping, and $\widetilde{\mathcal{O}}(\cdot)$ omits logarithmic terms. Finally, under the offline setting, we establish an upper bound on the suboptimality of the proposed algorithm for an arbitrary dataset with K trajectories. Our upper bound yields a sublinear $\widetilde{\mathcal{O}}(H^2\sqrt{d^3/K})$ rate as long as the dataset has sufficient coverage over the trajectory induced by the desired SNE.

Related work See §A for details.

Notation See §B for details.

2 Preliminaries

In this section, we introduce the formulation of the general-sum simultaneous-move Markov games, Stackelberg-Nash equilibrium, and the linear structure we use in this paper.

2.1 General-Sum Simultaneous-Move Markov Games

In this setting, two levels of hierarchy in decision making are considered: one leader l and N followers $\{f_i\}_{i\in[N]}$. Specifically, we define an episodic version of general-sum simultaneous-moves Markov game by the tuple $(\mathcal{S}, \mathcal{A}_l, \mathcal{A}_f = \{\mathcal{A}_{f_i}\}_{i\in[N]}, H, r_l, r_f = \{r_{f_i}\}_{i\in[N]}, \mathcal{P})$, where $\mathcal{S}$ is the state space, $\mathcal{A}_l$ and $\mathcal{A}_f$ are the sets of actions of the leader and the followers respectively, H is the number of steps in each episode, $r_l = \{r_{l,h} : \mathcal{S} \times \mathcal{A}_l \times \mathcal{A}_f \to [-1, 1]\}_{h=1}^H$ and $r_{f_i} = \{r_{f_i,h} : \mathcal{S} \times \mathcal{A}_l \times \mathcal{A}_f \to [-1, 1]\}_{h=1}^H$ are reward functions of the leader and the followers respectively, and $\mathcal{P} = \{\mathcal{P}_h : \mathcal{S} \times \mathcal{A}_l \times \mathcal{A}_f \times \mathcal{S} \to [0, 1]\}_{h=1}^H$ is a collection of transition kernels. Here $\mathcal{A}_l \times \mathcal{A}_f = \mathcal{A}_l \times \mathcal{A}_{f_1} \times \cdots \times \mathcal{A}_{f_N}$. Throughout this paper, we also let $\star$ be some element in $\{l, f_1, \cdots, f_N\}$. Moreover, for any $(h, x, a) \in [H] \times \mathcal{S} \times \mathcal{A}_l$ and $b = \{b_i \in \mathcal{A}_{f_i}\}_{i\in[N]}$, we use the shorthands $r_{\star,h}(x, a, b) = r_{\star,h}(x, a, b_1, \cdots, b_N)$ and $\mathcal{P}_h(\cdot \,|\, x, a, b) = \mathcal{P}_h(\cdot \,|\, x, a, b_1, \cdots, b_N)$.

Policy and Value Function. A stochastic policy $\pi = \{\pi_h : \mathcal{S} \to \Delta(\mathcal{A}_l)\}_{h=1}^H$ of the leader is a set of probability distributions over actions given the state. Meanwhile, a stochastic joint policy of the followers is defined by $\nu = \{\nu_{f_i}\}_{i\in[N]}$, where $\nu_{f_i} = \{\nu_{f_i,h} : \mathcal{S} \to \Delta(\mathcal{A}_{f_i})\}_{h=1}^H$. We use the notation $\pi_h(a \,|\, x)$ and $\nu_{f_i,h}(b_i \,|\, x)$ to denote the probability of taking action $a \in \mathcal{A}_l$ or $b_i \in \mathcal{A}_{f_i}$ for state x at step h under policy π, ν_{f_i} respectively. Throughout this paper, for any $\nu = \{\nu_{f_i}\}_{i\in[N]}$ and $b = \{b_i\}_{i\in[N]}$, we use the shorthand $\nu_h(b \,|\, x) = \nu_{f_1,h}(b_1 \,|\, x) \times \cdots \times \nu_{f_N,h}(b_N \,|\, x)$.

Given policies $(\pi, \nu = \{\nu_{f_i}\}_{i\in[N]})$, the action-value (Q) and state-value (V) functions for the leader and followers are defined by

$$Q_{\star,h}^{\pi,\nu}(x, a, b) = \mathbb{E}_{\pi,\nu,h,x,a,b}\left[\sum_{t=h}^{H} r_{\star,h}(x_t, a_t, b_t)\right], \quad V_{\star,h}^{\pi,\nu}(x) = \mathbb{E}_{a\sim\pi_h(\cdot\,|\,x), b\sim\nu_h(\cdot\,|\,x)} Q_{\star,h}^{\pi,\nu}(x, a, b), \tag{2.1}$$

where the expectation $\mathbb{E}_{\pi,\nu,h,x,a,b}$ is taken over state-action pairs induced by the policies $(\pi, \nu = \{\nu_{f_i}\}_{i\in[N]})$ and the transition probability, when initializing the process with the triplet $(s, a, b = \{b_i\}_{i\in[N]})$ at step h. For notational simplicity, when h, x, a, b are clear from the context, we omit h, x, a, b from $\mathbb{E}_{\pi,\nu,h,x,a,b}$. By the definition in (2.1), we have the Bellman equation

$$V_{\star,h}^{\pi,\nu} = \langle Q_{\star,h}^{\pi,\nu}, \pi_h \times \nu_h\rangle_{\mathcal{A}_l\times\mathcal{A}_f}, \quad Q_{\star,h}^{\pi,\nu} = r_{\star,h} + \mathbb{P}_h V_{\star,h+1}^{\pi,\nu}, \quad \forall \star \in \{l, f_1, \cdots, f_N\}, \tag{2.2}$$

where $\pi_h \times \nu_h$ represents $\pi_h \times \nu_{f_1,h} \times \cdots \times \nu_{f_N,h}$. Here $\mathbb{P}_h$ is the operator which is defined by

$$(\mathbb{P}_h f)(x, a, b) = \mathbb{E}[f(x') \,|\, x' \sim \mathcal{P}_h(x' \,|\, x, a, b)] \tag{2.3}$$

for any function $f : \mathcal{S} \to \mathbb{R}$ and $(x, a, b) \in \mathcal{S} \times \mathcal{A}_l \times \mathcal{A}_f$.

2.2 Stackelberg-Nash Equilibrium

Given a leader policy π, a Nash equilibrium (Nash, 2016) of the followers is a joint policy $\nu^* = \{\nu_{f_i}^*\}_{i\in[N]}$, such that for any $x \in \mathcal{S}$ and $(i, h) \in [N] \times [H]$

$$V_{f_i,h}^{\pi,\nu^*}(x) \geq V_{f_i,h}^{\pi,\nu_{f_i},\nu_{f_{-i}}^*}(x), \quad \forall \nu_{f_i}. \tag{2.4}$$

Here $-i$ represents all indices in $[N]$ except i. For each leader policy π, we denote the set of best-response policies of the followers by $\mathrm{BR}(\pi)$, which is defined by

$$\mathrm{BR}(\pi) = \{\nu = \{\nu_{f_i}\}_{i\in[N]} \,|\, \nu \text{ is the NE of the followers given the leader policy } \pi\}. \tag{2.5}$$

Given the best-response set $\mathrm{BR}(\pi)$, we denote $\nu^*(\pi)$ the worst-case responses, which break ties against favor of the leader [1]. Specifically, we define $\nu^*(\pi)$ by

$$\nu^*(\pi) = \{\nu \in \mathrm{BR}(\pi) \,|\, V_{l,h}^{\pi,\nu}(x) \leq V_{l,h}^{\pi,\nu'}(x), \forall x \in \mathcal{S}, h \in [H], \nu' \in \mathrm{BR}(\pi)\}. \tag{2.6}$$

The Stackelberg-Nash equilibrium for the leader is the "best response to the best response", that is,

$$\mathrm{SNE}_l = \{\pi \,|\, V_{l,h}^{\pi,\nu^*(\pi)}(x) \geq V_{l,h}^{\pi',\nu^*(\pi')}(x), \forall x \in \mathcal{S}, h \in [H], \pi'\} \tag{2.7}$$

A Stackelberg-Nash equilibrium of the general-sum game is a policy pair $(\pi^*, \nu^* = \{\nu_{f_i}^*\}_{i\in[N]})$ such that $\nu^* \in \nu^*(\pi^*)$ and $\pi^* \in \mathrm{SNE}_l$.

Our goal is to find the Stackelberg equilibrium: the leader's optimal strategy, assuming the followers play their best response (Nash equilibrium) to the leader. We study this challenging bilevel optimization problem in both the online setting (Section 3) and the offline setting (Section 4).

[1] This is also known as pessimistic tie breaking (Conitzer & Sandholm, 2006). We also remark that our subsequent analysis still holds for the optimistic setting (Breton et al., 1988; Bucarey et al., 2019a).

2.3 LEADER-CONTROLLER LINEAR MARKOV GAMES

Inspired by the linear MDP studied in Jin et al. (2020b) for the single-agent RL, we study the linear Markov games (Xie et al., 2020), where the transition dynamics are linear in a feature map. Specifically, there exists a feature map $\phi' : \mathcal{S} \times \mathcal{A}_l \times \mathcal{A}_f \to \mathbb{R}^d$ such that

$$\mathcal{P}_h(\cdot \,|\, x, a, b) = \langle \phi'(x, a, b), \mu_h(\cdot) \rangle$$

for any $(x, a, b) \in \mathcal{S} \times \mathcal{A}_l \times \mathcal{A}_f$ and $h \in [H]$. Here $\mu_h = (\mu_h^{(1)}, \mu_h^{(2)}, \cdots, \mu_h^{(d)})$ are d unknown signed measures over $\mathcal{S}$. Moreover, throughout this paper, we focus on the leader-controller game (Filar & Vrieze, 2012; Bucarey et al., 2019a), where the future state only depends on the current state and the leader's action, that is,

$$\mathcal{P}_h(\cdot \,|\, x, a, b) = \mathcal{P}_h(\cdot \,|\, x, a)$$

for any $(x, a, b) \in \mathcal{S} \times \mathcal{A}_l \times \mathcal{A}_f$ and $h \in [H]$. Hence, it is naturally to define leader-controller linear Markov games as follows.

Assumption 2.1. Markov game $(\mathcal{S}, \mathcal{A}_l, \mathcal{A}_f = \{\mathcal{A}_{f_i}\}_{i\in[N]}, H, r_l, r_f = \{r_{f_i}\}_{i\in[N]}, \mathcal{P})$ is a leader-controller linear Markov game if there exists a feature map $\phi : \mathcal{S} \times \mathcal{A}_l \to \mathbb{R}^d$ such that

$$\mathcal{P}_h(\cdot \,|\, x, a, b) = \langle \phi(x, a), \mu_h(\cdot) \rangle$$

for any $(x, a, b) \in \mathcal{S} \times \mathcal{A}_l \times \mathcal{A}_f$ and $h \in [H]$. Here $\mu_h = (\mu_h^{(1)}, \mu_h^{(2)}, \cdots, \mu_h^{(d)})$ are d unknown signed measures over $\mathcal{S}$. Without loss of generality, we assume that $\|\mu_h(\mathcal{S})\| \leq \sqrt{d}$ for all $h \in [H]$.

The linear Markov game above is an extension of linear MDP studied in Jin et al. (2020b) for the single-agent RL. Specifically, when the followers play fixed and known policies, the linear Markov games reduce to the linear MDP.

3 MAIN RESULTS FOR THE ONLINE SETTING

In this section, we study the online setting, where a central controller controls one leader l and N followers $\{f_i\}_{i\in[N]}$. Our goal is to learn a Stackelberg-Nash equilibrium. In what follows, we formally describe the setup and learning objectives, and then present our algorithm and provide theoretic guarantees.

3.1 SETUP AND LEARNING OBJECTIVE

We consider the setting where the reward functions r_l and $r_f = \{r_{f_i}\}_{i\in[N]}$ are revealed to the learner before the game. This is reasonable since in practice the reward functions are usually artificially designed. Moreover, we focus on the episodic setting. Specifically, a Markov game is played for K episodes, each of which consists of H timesteps. At the beginning of the k-th episode, the leader and followers determine their policies $(\pi^k, \nu^k = \{\nu_{f_i}^k\}_{i\in[N]})$, and a fixed initial state $x_1^k = x_1$ is chosen. Here we assume the fixed initial state just for ease of presentation, and our subsequent results can be generalized to the setting where x_1^k is picked from a fixed distribution. Then the game proceeds as follows. At each step $h \in [H]$, the leader and the followers observe state $x_h^k \in \mathcal{S}$ and pick their own actions $a_h^k \sim \pi_h^k(\cdot \,|\, x_h^k)$ and $b_h^k = \{b_{i,h}^k \sim \nu_{f_i,h}^k(\cdot \,|\, x_h^k)\}_{i\in[N]}$. Subsequently, the environment transitions to the next state $x_{h+1}^k \sim \mathcal{P}_h(\cdot \,|\, x_h^k, a_h^k, b_h^k)$. Each episode terminates after H timesteps.

Learning Objective. By the definition in (2.5), given a leader's policy, the best response for the followers is the Nash equilibrium of followers' game induced by this leader's policy. Recall the definition of Nash equilibrium in (2.4), for any policies $(\pi, \nu = \{\nu_{f_i}\}_{i\in[N]})$, it is natural to define the following objective to measure the suboptimality of ν_{f_i}:

$$\text{SubOpt}_{f_i}(x) = V_{f_i,1}^{\pi,\nu^*(\pi)}(x) - V_{f_i,1}^{\pi,\nu_{f_i},\nu_{-i}^*(\pi)}(x).$$

Meanwhile, we evaluate the performance of the leader's policy π by the following suboptimality gap:

$$\text{SubOpt}_l(x) = V_{l,1}^{\pi^*,\nu^*}(x) - V_{l,1}^{\pi,\nu^*(\pi)}(x).$$

Putting these two suboptimality gaps together, we formally define the regret as follows.

Definition 3.1 (Regret)**.** Let $(\pi^k, \nu^k = \{\nu_{f_i}^k\}_{i\in[N]})$ denote the policies executed by the algorithm in the k-th episode. After a total of K episodes, the regret is defined as

$$\text{Regret}(K) = \underbrace{\sum_{k=1}^{K} V_{l,1}^{\pi^*,\nu^*}(x_1^k) - V_{l,1}^{\pi^k,\nu^*(\pi^k)}(x_1^k)}_{\text{Regret}_l(K)} + \underbrace{\sum_{i=1}^{N}\sum_{k=1}^{K} V_{f_i,1}^{\pi^k,\nu^*(\pi^k)}(x_1^k) - V_{f_i,1}^{\pi^k,\nu_{f_i}^k,\nu_{f_{-i}}^*(\pi^k)}(x_1^k)}_{\text{Regret}_f(K)}. \tag{3.1}$$

The goal is to design algorithms with regret that is sublinear in K, and polynomial in d, H. Here K is the number of episodes, d is the dimension of the feature map ϕ, and H is the episode horizon.

3.2 Algorithm

We now present our algorithm, Optimistic Value Iteration to Find Stackelberg-Nash Equilibrium (OVI-SNE), which is given in Algorithm 1.

At a high level, in each episode, our algorithm first construct the policies for all players through backward induction with respect to the timestep h (line 4-11), and then execute the policies to play the game (line 12-16).

In detail, at h-th step of k-th episode, OVI-SNE estimates leader's Q-function based on the $(k-1)$ historical trajectories. Inspired by previous optimistic least square value iteration (LSVI) algorithms (Jin et al., 2020b), for any $h \in [H]$, we estimate the linear coefficients by solving the following ridge regression problem:

$$\begin{aligned} w_h^k &\leftarrow \operatorname*{argmin}_{w\in\mathbb{R}^d} \sum_{\tau=1}^{k-1} [V_{h+1}^k(x_{h+1}^\tau) - \phi(x_h^\tau, a_h^\tau)^\top w]^2 + \|w\|^2, \\ &\text{where } V_{h+1}^k(\cdot) = \langle Q_{h+1}^k(\cdot,\cdot,\cdot), \pi_{h+1}^k(\cdot\,|\,\cdot) \times \nu_{h+1}^k(\cdot\,|\,\cdot)\rangle_{\mathcal{A}_l\times\mathcal{A}_f}. \end{aligned} \tag{3.2}$$

By solving the ridge regression problem in (3.2), we have

$$\begin{aligned} w_h^k &= (\Lambda_h^k)^{-1}\Big(\sum_{\tau=1}^{k-1} \phi(x_h^\tau, a_h^\tau)\cdot V_{h+1}^k(x_{h+1}^\tau)\Big), \\ &\text{where } \Lambda_h^k = \sum_{\tau=1}^{k-1} \phi(x_h^\tau, a_h^\tau)\phi(x_h^\tau, a_h^\tau)^\top + I. \end{aligned} \tag{3.3}$$

To encourage exploration, we additionally adds a bonus function to estimate the leader's Q-function:

$$\begin{aligned} Q_h^k(\cdot,\cdot,\cdot) &\leftarrow r_{l,h}(\cdot,\cdot,\cdot) + \Pi_{H-h}\{\phi(\cdot,\cdot)^\top w_h^k + \Gamma_h^k(\cdot,\cdot)\}, \\ &\text{where } \Gamma_h^k(\cdot,\cdot) = \beta\cdot\sqrt{\phi(\cdot,\cdot)^\top(\Lambda_h^k)^{-1}\phi(\cdot,\cdot)}. \end{aligned} \tag{3.4}$$

Here $\Gamma_h^k : \mathcal{S}\times\mathcal{A}_l \to \mathbb{R}$ is a bonus function and $\beta > 0$ is a parameter which will be specified later. This form of bonus function is common in the literature of linear bandits (Lattimore & Szepesvári, 2020) and linear MDPs (Jin et al., 2020b).

Then, we construct policies for the leader and followers by the subroutine ϵ-SNE (Algorithm 2). Specifically, let $\mathcal{Q}_h^k$ be the class of functions $Q : \mathcal{S}\times\mathcal{A}_l\times\mathcal{A}_f \to \mathbb{R}$ that takes form

$$Q(\cdot,\cdot,\cdot) = r_{l,h}(\cdot,\cdot,\cdot) + \Pi_{H-h}\big\{\phi(\cdot,\cdot)^\top w + \beta\cdot\big(\phi(\cdot,\cdot)^\top\Lambda^{-1}\phi(\cdot,\cdot)\big)^{1/2}\big\}, \tag{3.5}$$

where the parameters $(w,\Lambda) \in \mathbb{R}^d\times\mathbb{R}^{d\times d}$ satisfy $\|w\| \le H\sqrt{dk}$ and $\lambda_{\min}(\Lambda) \ge 1$. Moreover, let $\mathcal{Q}_{h,\epsilon}^k$ be a fixed ϵ-covering of $\mathcal{Q}_h^k$ with respect to the ℓ_∞ norm. By Lemma C.10, we have $Q_h^k \in \mathcal{Q}_h^k$, which allows us to pick a $\widetilde{Q} \in \mathcal{Q}_{h,\epsilon}^k$ such that $\|\widetilde{Q} - Q_h^k\|_\infty \le \epsilon$ and calculate policies by

$$(\pi_h^k(\cdot\,|\,x), \{\nu_{f_i,h}^k(\cdot\,|\,x)\}_{i\in[N]}) \leftarrow \text{SNE}(\widetilde{Q}(x,\cdot,\cdot), \{r_{f_i,h}(x,\cdot,\cdot)\}_{i\in[N]}),\ \forall x. \tag{3.6}$$

When there is only one follower, such a problem can be transformed to a linear programming (LP) problem (Conitzer & Sandholm, 2006; Von Stengel & Zamir, 2010), and thus can be solved efficiently. For the multi-follower case, however, solving such a matrix game in general is hard

(Conitzer & Sandholm, 2006; Basilico et al., 2017a;b; Coniglio et al., 2020). Given this computational hardness, we focus on the sample complexity and explicitly assume access to the following computational oracle:

Assumption 3.2. We assume access to an oracle that implements Line 3 of Algorithm 2.

Now we explain the motivation for using the subroutine ϵ-SNE to construct policies instead of solving the matrix games with payoff matrices $(Q_h^k(x,\cdot,\cdot), \{r_{f_i,h}(x,\cdot,\cdot)\}_{i\in[N]})$ directly. By the definition of Q_h^k in (3.4), we know Q_h^k relies on the previous data via the estimated value function V_{h+1}^k and feature maps $\{\phi(x_h^\tau, a_h^\tau, b_h^\tau)\}_{\tau=1}^{k-1}$. Similar to the analysis for linear MDPs (Jin et al., 2020b), we need to use a covering argument to establish uniform concentration bounds for all value V_{h+1}^k. Jin et al. (2020b) directly constructs an ϵ-net for the value functions and establishes a polynomial log-covering number for this ϵ-net. This analysis, however, relies on that the policies executed by the players are greedy (deterministic), which is not valid for our setting. To overcome this technical issue, we construct an ϵ-net for Q-functions and solve an approximate matrix game. Fortunately, by choosing a small enough ϵ, we can handle the errors caused by this approximation. See §C for more details. Moreover, as shown in Xie et al. (2020), this subroutine can be implemented efficiently without explicitly computing the exponentially large ϵ-net.

Finally, the leader and the followers play the game according to the obtained policies.

Algorithm 1 Optimistic Value Iteration to Find Stackelberg-Nash Equilibria

1: Initialize $V_{l,H+1}(\cdot) = V_{f,H+1}(\cdot) = 0$.
2: **for** $k = 1, 2, \cdots, K$ **do**
3: Receive initial state x_1^k.
4: **for** step $h = H, H-1, \cdots, 1$ **do**
5: $\Lambda_h^k \leftarrow \sum_{\tau=1}^{k-1} \phi(x_h^\tau, a_h^\tau)\phi(x_h^\tau, a_h^\tau)^\top + I$.
6: $w_h^k \leftarrow (\Lambda_h^k)^{-1} \sum_{\tau=1}^{k-1} \phi(x_h^\tau, a_h^\tau) \cdot V_{h+1}^k(x_{h+1}^\tau)$.
7: $\Gamma_h^k(\cdot,\cdot) \leftarrow \beta \cdot (\phi(\cdot,\cdot)^\top (\Lambda_h^k)^{-1} \phi(\cdot,\cdot))^{1/2}$.
8: $Q_h^k(\cdot,\cdot,\cdot) \leftarrow r_{l,h}(\cdot,\cdot,\cdot) + \Pi_{H-h}\{\phi(\cdot,\cdot)^\top w_h^k + \Gamma_h^k(\cdot,\cdot)\}$.
9: $(\pi_h^k(\cdot\,|\,x), \{\nu_{f_i,h}^k(\cdot\,|\,x)\}_{i\in[N]}) \leftarrow \epsilon\text{-SNE}(Q_h^k(x,\cdot,\cdot), \{r_{f_i,h}(x,\cdot,\cdot)\}_{i\in[N]}),\ \forall x$. (Alg. 2)
10: $V_h^k(x) \leftarrow \mathbb{E}_{a\sim\pi_h^k(\cdot\,|\,x), b_1\sim\nu_{f_1,h}^k(\cdot\,|\,x), \cdots, b_N\sim\nu_{f_N,h}^k(\cdot\,|\,x)} Q_h^k(x, a, b_1, \cdots, b_N),\ \forall x$.
11: **end for**
12: **for** $h = 1, 2, \cdot, H$ **do**
13: Sample $a_h^k \sim \pi_h^k(\cdot\,|\,x_h^k), b_{1,h}^k \sim \nu_{f_1,h}^k(\cdot\,|\,x_h^k), \cdots, b_{N,h}^k \sim \nu_{f_N,h}^k(\cdot\,|\,x_h^k)$.
14: Leader takes action a_h^k; Followers take actions $b_h^k = \{b_{i,h}^k\}_{i\in[N]}$.
15: Observe next state x_{h+1}^k.
16: **end for**
17: **end for**

Algorithm 2 ϵ-SNE

1: **Input:** Q_h^k, x, and parameter ϵ.
2: Select $\widetilde{Q}$ from $\mathcal{Q}_{h,\epsilon}^k$ satisfying $\|\widetilde{Q} - Q_h^k\|_\infty \le \epsilon$.
3: For the input state x, let $(\pi_h^k(\cdot\,|\,x), \{\nu_{f_i,h}^k(\cdot\,|\,x)\}_{i\in[N]})$ be the Stackelberg-Nash equilibrium for the matrix game with payoff matrices $(\widetilde{Q}(x,\cdot,\cdot), \{r_{f_i,h}(x,\cdot,\cdot)\}_{i\in[N]})$.
4: **Output:** $(\pi_h^k(\cdot\,|\,x), \{\nu_{f_i,h}^k(\cdot\,|\,x)\}_{i\in[N]})$.

3.3 Theoretical Results

Our main theoretical result is the following bound on the regret incurred by Algorithm 1. Recall that the regret is defined in Definition 3.1 and $T = KH$ is the total number of timesteps.

Theorem 3.3. Under Assumptions 2.1 and 3.2, there exists an absolute constant $C > 0$ such that, for any fixed $p \in (0,1)$, by setting $\beta = C \cdot dH\sqrt{\iota}$ with $\iota = \log(2dT/p)$ in Line 7 of Algorithm 1 and $\epsilon = \frac{1}{KH}$ in Algorithm 2, then with probability at least $1-p$, the regret incurred by OVI-SNE

satisfies that

$$\text{Regret}(K) \leq \mathcal{O}(\sqrt{d^3H^3T\iota^2}).$$

Proof. See §C for a detailed proof. □

Learning Stackelberg Equilibria. When there is only one follower, Stackelberg-Nash equilibrium reduces to the Stackelberg equilibrium (Simaan & Cruz, 1973; Conitzer & Sandholm, 2006; Bai et al., 2021). Thus, we partly answer the open problem in Bai et al. (2021) on how to learn Stackelberg equilibria in (leader-controller) Markov games.

Optimality of the Bound. Assuming that the action of the follower won't affect the transition kernel and reward function, the linear Markov games reduces to the linear MDP (Jin et al., 2020b). Meanwhile, the lower bound established in Azar et al. (2017); Jin et al. (2018) for tabular MDPs and the lower bound established in Lattimore & Szepesvári (2020) for linear bandits directly imply a lower bound $\Omega(dH\sqrt{T})$ for the linear MDPs, which further yields a lower bound $\Omega(dH\sqrt{T})$ for our setting. Ignoring the logarithmic factors, there is only a gap of $\sqrt{dH}$ between this lower bound and our upper bound. We also point out that, by using the "Bernstein-type" bonus (Azar et al., 2017; Jin et al., 2018; Zhou et al., 2020), we can improve our upper bound by a factor of $\sqrt{H}$. Here we don't apply this technique for the clarity of the analysis.

Misspecification. For ease of presentation, we assume the Markov games are leader-controller in Assumption 2.1. When the transitions do not ideally satisfy the leader-controller assumption, we can potentially consider cases that transitions satisfy, for instance, $|\mathcal{P}_h(\cdot \,|\, x, a, b) - \mathcal{P}_h(\cdot \,|\, x, a)\|_\infty \leq \varrho$ for any $(h, x, a, b) \in [H] \times \mathcal{S} \times \mathcal{A}_l \times \mathcal{A}_f$, Here ϱ is the misspecification error. We can still follow the above method to tackle the misspecified cases. However, because of the misspecification error cumulated during T steps, an extra term $\mathcal{O}(\varrho T)$ will appear in the final result. In particular, When ϱ is small, that is the Markov games have approximately leader-controller transitions, the extra term $\mathcal{O}(\varrho T)$ should be small, which further indicates that we can find SNEs efficiently in some misspecified general-sum Markov games.

Unknown Reward Setting. To relax the assumption that the reward is known, in this subsection, we consider the case where the reward functions are unknown. We focus on the tabular case for simplicity, and the extension to linear case is left as future work. We assume that $S = |\mathcal{S}|$, $A_l = |\mathcal{A}_l|$ and $A_f = |\mathcal{A}_f| = |\mathcal{A}_{f_1} \times \cdots \times \mathcal{A}_{f_N}|$. For simplicity, we use the shorthand $V_\star^{\pi,\nu} = V_{\star,1}^{\pi,\nu}(x_1)$, where $x_1 \in \mathcal{S}$ is the fixed initial state.

At a high level, we first conduct a reward-free exploration algorithm (Algorithm 4 in §D), a variant of Reward-Free RL-Explore algorithm in Jin et al. (2020a), to obtain estimated reward functions $\{\widehat{r}_l, \widehat{r}_{f_1}, \cdots \widehat{r}_{f_N}\}$. As asserted before, we can use Algorithm 1, to find the SNE with respect to the *known* estimated reward functions $\{\widehat{r}_l, \widehat{r}_{f_1}, \cdots \widehat{r}_{f_N}\}$. Hence, we can obtain the approximate SNE if the value functions of estimated value functions are good approximation of the true value functions. Fortunately, we have the following lemma to guarantee it.

Lemma 3.4. Fix $\varepsilon, p > 0$. If we set $K_0 \geq \Omega(H^7S^4A_l/\varepsilon)$ and $K \geq \Omega(H^3S^2A_lA_f/\varepsilon^2)$ in Algorithm 4 (cf. §D), then we have the empirical rewards $\{\widehat{r}_l, \widehat{r}_{f_1}, \cdots \widehat{r}_{f_N}\}$ and corresponding value functions $(\widehat{V}_l, \widehat{V}_{f_1}, \cdots, \widehat{V}_{f_N})$ satisfying that

$$\sup_{\pi,\nu} |\widehat{V}_\star^{\pi,\nu} - V_\star^{\pi,\nu}| \leq \varepsilon$$

with probability at least $1 - p$. Here $\Omega(\cdot)$ hides some logarithmic factors.

Proof. This lemma is a simple extension of Lemma D.1 in Bai et al. (2021). They focus on the MDP setting and we consider the more complex leader-controller Markov games. The detailed proof is given in §E. □

Lemma 3.4 states that we can obtain estimated reward functions and the associated value functions is an ε-approximation of the true value functions, which further implies that the SNE with respect to the estimated reward functions is a good approximation of the SNE in the original problem. We also remark that if we consider the Markov games with only one follower and aim to find the Stackelberg equilibria, we can provide a more refined analysis. See §F for more details.

4 MAIN RESULTS FOR THE OFFLINE SETTING

In this section, we study the offline setting, where the central controller aims to find a Stackelberg-Nash equilibrium by an offline dataset. Below we describe the setup and learning objective, followed by our algorithm and theoretical results.

4.1 SETUP AND LEARNING OBJECTIVE

We study the offline setting, where the learner has access to the reward functions $(r_l, r_f = \{r_{f_i}\}_{i=1}^N)$ and a dataset $\mathcal{D} = \{(x_h^\tau, a_h^\tau, b_h^\tau = \{b_{i,h}^\tau\}_{i=1}^N)\}_{\tau,h=1}^{K,H}$, which is collected a priori by some experimenter. Then we make a minimal assumption for the offline dataset.

Assumption 4.1 (Compliance of Dataset)**.** We assume that the dataset $\mathcal{D}$ is compliant with the underlying Markov game $(\mathcal{S}, \mathcal{A}_l, \mathcal{A}_f, H, r_l, r_f, \mathcal{P})$, that is, for any $x' \in \mathcal{S}$ at step $h \in [H]$ of each trajectory $\tau \in [K]$,

$$P_{\mathcal{D}}(x_{h+1}^\tau = x' \,|\, \{x_h^j, a_h^j, b_h^j, x_{h+1}^j\}_{j=1}^{\tau-1} \cup \{x_h^\tau, a_h^\tau, b_h^\tau\}) = P(x_{h+1} = x' \,|\, x_h = x_h^\tau, a_h = a_h^\tau).$$

Here the probability on the left-hand side is with respect to the joint distribution over dataset $\mathcal{D}$ and the probability on the right-hand side is with respect to the underlying Markov game.

Assumption 4.1 is adopted from Jin et al. (2020c), which indicates the Markov property of the dataset $\mathcal{D}$ and that x_{h+1}^τ is generated by the underlying Markov game conditioned on $(x_h^\tau, a_h^\tau, b_h^\tau)$. As a special case, Assumption 4.1 holds when the experimenter follows fixed behavior policies. More generally, Assumption 4.1 allows the experimenter to choose actions a_h^τ and b_h^τ arbitrarily, even in an adaptive or adversarial manner. In particular, we can assume that a_h^τ and b_h^τ are interdependent across each trajectory $\tau \in [K]$. For instance, the experimenter can sequentially improve the behavior policy using any online algorithm for Markov games.

Learning Objective. Similar to the online setting, we define the following performance metric

$$\text{SubOpt}(\pi, \nu, x) = \underbrace{V_{l,1}^{\pi^*,\nu^*}(x) - V_{l,1}^{\pi,\nu^*(\pi)}(x)}_{\text{SubOpt}_l} + \underbrace{\sum_{i=1}^N [V_{f_i,1}^{\pi,\nu^*(\pi)}(x) - V_{f_i,1}^{\pi,\nu_{f_i},\nu_{f_{-i}}^*(\pi)}(x)]}_{\text{SubOpt}_f}, \tag{4.1}$$

which evaluates the suboptimality of policies $(\pi, \nu = \{\nu_{f_i}\}_{i=1}^N)$ given the initial state $x \in \mathcal{S}$.

4.2 ALGORITHM AND THEORETICAL RESULTS

As is known to us, the key challenge of online setting is the the tradeoff between exploration and exploration. In the online setting. by following the "optimism in the face of uncertainty" principle (Sutton & Barto, 2018), we use bonus functions to incentivize exploration and thus achieve sample-efficient. This intrinsic challenge of online setting disappears in the offline setting because we do not need exploration any more. But another challenge arises: we only have access to the limited data. To tackle this challenge, we need add some penalty functions to achieve robustness against the uncertainty due to the finite data. This is also known as pessimism (Yu et al., 2020; Jin et al., 2020c; Liu et al., 2020b; Buckman et al., 2020; Kidambi et al., 2020; Kumar et al., 2020; Rashidinejad et al., 2021). Here we simply flip the sign of bonus functions defined in (3.4) to serve as penalty functions. See Algorithm 3 for details.

Suppose that $(\widehat{\pi}, \widehat{\nu})$ are the output policies of Algorithm 3. Then we evaluate the performance of $(\widehat{\pi}, \widehat{\nu})$ by establishing an upper bound for the optimality gap defined in (4.1).

Theorem 4.2. Under Assumptions 2.1, 3.2, and 4.1, there exists an absolute constant $C > 0$ such that, for any fixed $p \in (0, 1)$, by setting $\beta' = C \cdot dH\sqrt{\log(2dHK/p)}$ in Line 6 of Algorithm 3 and $\epsilon = \frac{d}{KH}$ in Algorithm 2, then with probability at least $1 - p$, we have

$$\text{SubOpt}(\widehat{\pi}, \widehat{\nu}, x) \le 3\beta' \sum_{h=1}^H \mathbb{E}_{\pi^*,x}\big[\big(\phi(s_h, a_h)^\top (\Lambda_h)^{-1} \phi(s_h, a_h)\big)^{1/2}\big], \tag{4.2}$$

where $\mathbb{E}_{\pi^*,x}$ is taken with respect to the trajectory incurred by π^* in the underlying leader-controller Markov game when initializing the progress at x. Here Λ_h is defined in Line 4 of Algorithm 3.

Proof. See §G for a detailed proof. □

Algorithm 3 Pessimistic Value Iteration to Find Stackelberg-Nash Equilibria

1: **Input:** $\mathcal{D} = \{x_h^\tau, a_h^\tau, b_h^\tau = \{b_{i,h}^\tau\}_{i\in[N]}\}_{\tau,h=1}^{K,H}$ and reward functions $\{r_l, r_f = \{r_{f_i}\}_{i\in[N]}\}$.
2: Initialize $\widehat{V}_{H+1}(\cdot) = 0$.
3: **for** step $h = H, H-1, \cdots, 1$ **do**
4: $\quad \Lambda_h \leftarrow \sum_{\tau=1}^K \phi(x_h^\tau, a_h^\tau)\phi(x_h^\tau, a_h^\tau)^\top + I$.
5: $\quad w_h \leftarrow (\Lambda_h)^{-1} \sum_{\tau=1}^K \phi(x_h^\tau, a_h^\tau) \cdot \widehat{V}_{h+1}(x_{h+1}^\tau)$.
6: $\quad \Gamma_h(\cdot,\cdot) \leftarrow \beta' \cdot (\phi(\cdot,\cdot)^\top (\Lambda_h)^{-1} \phi(\cdot,\cdot))^{1/2}$.
7: $\quad \widehat{Q}_h(\cdot,\cdot,\cdot) \leftarrow r_{l,h}(\cdot,\cdot,\cdot) + \Pi_{H-h}\{\phi(\cdot,\cdot)^\top w_h - \Gamma_h(\cdot,\cdot)\}$.
8: $\quad (\widehat{\pi}_h(\cdot\,|\,x), \{\widehat{\nu}_{f_i,h}(\cdot\,|\,x)\}_{i\in[N]}) \leftarrow \epsilon\text{-SNE}(\widehat{Q}_h(x,\cdot,\cdot), \{r_{f_i,h}(x,\cdot,\cdot)\}_{i\in[N]}),\ \forall x$. (Alg. 2)
9: $\quad \widehat{V}_h(x) \leftarrow \mathbb{E}_{a\sim\widehat{\pi}_h(\cdot\,|\,x), b_1\sim\widehat{\nu}_{f_1,h}(\cdot\,|\,x), \cdots, b_N\sim\widehat{\nu}_{f_N,h}(\cdot\,|\,x)} \widehat{Q}_h(x, a, b_1, \cdots, b_N),\ \forall x$.
10: **end for**
11: **Output:** $(\widehat{\pi} = \{\widehat{\pi}_h\}_{h=1}^H, \widehat{\nu} = \{\widehat{\nu}_{f_i} = \{\nu_{f_i,h}\}_{h=1}^H\}_{i=1}^N)$.

Minimal Assumption Requirement: Theorem 4.2 only relies on the compliance of the dataset with linear Markov games. Compared with existing literature on offline RL (Bertsekas & Tsitsiklis, 1996; Antos et al., 2007; 2008; Munos & Szepesvári, 2008; Farahmand et al., 2010; 2016; Scherrer et al., 2015; Liu et al., 2018; Chen & Jiang, 2019; Fan et al., 2020; Xie & Jiang, 2020), we impose no restrictions on the coverage of the dataset. Meanwhile, we need no assumption on the affinity between $(\widehat{\pi}, \widehat{\nu})$ and the behavior policies that induce the dataset, which is often employed as a regularizer (Fujimoto et al., 2019; Laroche et al., 2019; Jaques et al., 2019; Wu et al., 2019; Kumar et al., 2019; Wang et al., 2020; Siegel et al., 2020; Nair et al., 2020; Liu et al., 2020b).

Dataset with Sufficient Coverage: In what follows, we specialize Theorem 4.2 to the setting where we assume the dataset with good "coverage". Note that Λ_h is determined by the offline dataset $\mathcal{D}$ and acts as a fixed matrix in the expectation, that is, the expectation in (4.2) is only taken with the trajectory induced by π^*. As proofed in the following theorem, when the trajectory induced by π^* is "covered" by the dataset $\mathcal{D}$ sufficiently well, we can establish that the suboptimality incurred by Algorithm 3 diminishes at rate of $\widetilde{\mathcal{O}}(1/\sqrt{K})$.

Corollary 4.3. Suppose it holds with probability at least $1 - p/2$ that

$$\Lambda_h \succeq I + c \cdot K \cdot \mathbb{E}_{\pi^*,x}[\phi(s_h, a_h)\phi(s_h, a_h)^\top]$$

for all $(x, h) \in \mathcal{S} \times [H]$. Here $c > 0$ is an absolute constant and $\mathbb{E}_{\pi^*,x}$ is taken with respect to the trajectory incurred by π^* in the underlying leader-controller Markov game when initializing the progress at x. Under Assumptions 2.1, 3.2 and 4.1, there exists an absolute constant $C > 0$ such that, for any fixed $p \in (0, 1)$, by setting $\beta' = C \cdot dH\sqrt{\log(4dHK/p)}$ in Line 6 of Algorithm 3 and $\epsilon = \frac{d}{KH}$ in Algorithm 2, then it holds with probability at least $1 - p$ that

$$\text{SubOpt}(\widehat{\pi}, \widehat{\nu}, x) \le \bar{C} \cdot d^{3/2} H^2 \sqrt{\log(4dHK/p)/K}$$

for all $x \in \mathcal{S}$. Here $\bar{C}$ is another absolute constant that only depends on c and C.

Proof. See §H for a detailed proof. □

Note that, unlike the previous literature (Antos et al., 2007; Munos & Szepesvári, 2008; Farahmand et al., 2010; 2016; Scherrer et al., 2015; Liu et al., 2018; Chen & Jiang, 2019; Fan et al., 2020; Xie & Jiang, 2020) which relies on the "uniform coverage" assumption, Corollary 4.3 only assumes that the dataset has a good coverage of the trajectory incurred by the policy π^*.

Optimality of the Bound: Assuming the dummy followers, that is, the actions taken by the followers won't affect the reward functions and transition kernels, the Markov games reduces to the linear MDP (Jin et al., 2020b). Together with the information-theoretic lower bound $\Omega(\sum_{h=1}^H \mathbb{E}_{\pi^*,x}[(\phi(s_h, a_h)^\top (\Lambda_h)^{-1} \phi(s_h, a_h))^{1/2}])$ established in Jin et al. (2020c) for linear MDPs, we immediately obtain the same lower bound for our setting. In particular, our upper bound established in Theorem 4.2 matches this lower bound up to β' and absolute constants and thus implies that our algorithm is nearly minimax optimal.

REFERENCES

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *NIPS*, volume 11, pp. 2312–2320, 2011.

Alekh Agarwal, Sham Kakade, and Lin F Yang. Model-based reinforcement learning with a generative model is minimax optimal. In *Conference on Learning Theory*, pp. 67–83. PMLR, 2020.

Forest Agostinelli, Stephen McAleer, Alexander Shmakov, and Pierre Baldi. Solving the rubik's cube with deep reinforcement learning and search. *Nature Machine Intelligence*, 1(8):356–363, 2019.

Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, et al. Solving rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113*, 2019.

András Antos, Rémi Munos, and Csaba Szepesvári. Fitted q-iteration in continuous action-space mdps. 2007.

András Antos, Csaba Szepesvári, and Rémi Munos. Learning near-optimal policies with bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71(1):89–129, 2008.

Alex Ayoub, Zeyu Jia, Csaba Szepesvari, Mengdi Wang, and Lin Yang. Model-based reinforcement learning with value-targeted regression. In *International Conference on Machine Learning*, pp. 463–474. PMLR, 2020.

Mohammad Gheshlaghi Azar, Rémi Munos, and Hilbert J Kappen. Minimax PAC bounds on the sample complexity of reinforcement learning with a generative model. *Machine learning*, 91(3): 325–349, 2013.

Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*, pp. 263–272. PMLR, 2017.

Yu Bai and Chi Jin. Provable self-play algorithms for competitive reinforcement learning. In *International Conference on Machine Learning*, pp. 551–560. PMLR, 2020.

Yu Bai, Chi Jin, and Tiancheng Yu. Near-optimal reinforcement learning with self-play. *arXiv preprint arXiv:2006.12007*, 2020.

Yu Bai, Chi Jin, Huan Wang, and Caiming Xiong. Sample-efficient learning of stackelberg equilibria in general-sum games. *arXiv preprint arXiv:2102.11494*, 2021.

Tamer Başar and Geert Jan Olsder. *Dynamic noncooperative game theory*. SIAM, 1998.

Nicola Basilico, Stefano Coniglio, and Nicola Gatti. Methods for finding leader–follower equilibria with multiple followers. *arXiv preprint arXiv:1707.02174*, 2017a.

Nicola Basilico, Stefano Coniglio, Nicola Gatti, and Alberto Marchesi. Bilevel programming approaches to the computation of optimistic and pessimistic single-leader-multi-follower equilibria. In *SEA*, volume 75, pp. 1–14. Schloss Dagstuhl-Leibniz-Zentrum fur Informatik GmbH, Dagstuhl Publishing, 2017b.

Dimitri P Bertsekas and John N Tsitsiklis. *Neuro-dynamic programming*. Athena Scientific, 1996.

Avrim Blum, Nika Haghtalab, and Ariel D Procaccia. Learning optimal commitment to overcome insecurity. 2014.

Michele Breton, Abderrahmane Alj, and Alain Haurie. Sequential stackelberg equilibria in two-person games. *Journal of Optimization Theory and Applications*, 59(1):71–97, 1988.

Víctor Bucarey, Eugenio Della Vecchia, Alain Jean-Marie, and Fernando Ordóñez. *Stationary Strong Stackelberg Equilibrium in Discounted Stochastic Games*. PhD thesis, INRIA, 2019a.

Víctor Bucarey, Alain Jean-Marie, Eugenio Della Vecchia, and Fernando Ordóñez. On the value iteration method for dynamic strong stackelberg equilibria. In *ROADEF 2019-20ème congrès annuel de la société Française de Recherche Opérationnelle et d'Aide à la Décision*, 2019b.

Jacob Buckman, Carles Gelada, and Marc G Bellemare. The importance of pessimism in fixed-dataset policy optimization. *arXiv preprint arXiv:2009.06799*, 2020.

Lucian Busoniu, Robert Babuska, and Bart De Schutter. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(2):156–172, 2008.

Qi Cai, Zhuoran Yang, Chi Jin, and Zhaoran Wang. Provably efficient exploration in policy optimization. In *International Conference on Machine Learning*, pp. 1283–1294. PMLR, 2020.

Jinglin Chen and Nan Jiang. Information-theoretic considerations in batch reinforcement learning. In *International Conference on Machine Learning*, pp. 1042–1051. PMLR, 2019.

Tianyi Chen, Yuejiao Sun, and Wotao Yin. A single-timescale stochastic bilevel optimization method. *arXiv preprint arXiv:2102.04671*, 2021a.

Zhuoqun Chen, Yangyang Liu, Bo Zhou, and Meixia Tao. Caching incentive design in wireless d2d networks: A stackelberg game approach. In *2016 IEEE International Conference on Communications (ICC)*, pp. 1–6. IEEE, 2016.

Zixiang Chen, Dongruo Zhou, and Quanquan Gu. Almost optimal algorithms for two-player markov games with linear function approximation. *arXiv preprint arXiv:2102.07404*, 2021b.

Stefano Coniglio, Nicola Gatti, and Alberto Marchesi. Computing a pessimistic stackelberg equilibrium with multiple followers: The mixed-pure case. *Algorithmica*, 82(5):1189–1238, 2020.

Vincent Conitzer and Tuomas Sandholm. Complexity of mechanism design. *arXiv preprint cs/0205075*, 2002.

Vincent Conitzer and Tuomas Sandholm. Computing the optimal strategy to commit to. In *Proceedings of the 7th ACM conference on Electronic commerce*, pp. 82–90, 2006.

Qiwen Cui and Lin F Yang. Minimax sample complexity for turn-based stochastic game. *arXiv preprint arXiv:2011.14267*, 2020.

Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. 2008.

Constantinos Daskalakis, Dylan J Foster, and Noah Golowich. Independent policy gradient methods for competitive reinforcement learning. *arXiv preprint arXiv:2101.04233*, 2021.

Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *International conference on machine learning*, pp. 1329–1338. PMLR, 2016.

Yonathan Efroni, Lior Shani, Aviv Rosenberg, and Shie Mannor. Optimistic policy optimization with bandit feedback. *arXiv preprint arXiv:2002.08243*, 2020.

Jianqing Fan, Zhaoran Wang, Yuchen Xie, and Zhuoran Yang. A theoretical analysis of deep q-learning. In *Learning for Dynamics and Control*, pp. 486–489. PMLR, 2020.

Amir Massoud Farahmand, Rémi Munos, and Csaba Szepesvári. Error propagation for approximate policy and value iteration. In *Advances in Neural Information Processing Systems*, 2010.

Amir-massoud Farahmand, Mohammad Ghavamzadeh, Csaba Szepesvári, and Shie Mannor. Regularized policy iteration with nonparametric function spaces. *The Journal of Machine Learning Research*, 17(1):4809–4874, 2016.

Tanner Fiez, Benjamin Chasnov, and Lillian J Ratliff. Convergence of learning dynamics in stackelberg games. *arXiv preprint arXiv:1906.01217*, 2019.

Jerzy Filar and Koos Vrieze. *Competitive Markov decision processes*. Springer Science & Business Media, 2012.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International Conference on Machine Learning*, pp. 2052–2062. PMLR, 2019.

Dinesh Garg and Yadati Narahari. Design of incentive compatible mechanisms for stackelberg problems. In *International Workshop on Internet and Network Economics*, pp. 718–727. Springer, 2005.

Saeed Ghadimi and Mengdi Wang. Approximation methods for bilevel programming. *arXiv preprint arXiv:1802.02246*, 2018.

Amy Greenwald, Keith Hall, and Roberto Serrano. Correlated q-learning. In *ICML*, volume 3, pp. 242–249, 2003.

Thomas Dueholm Hansen, Peter Bro Miltersen, and Uri Zwick. Strategy iteration is strongly polynomial for 2-player turn-based stochastic games with a constant discount factor. *Journal of the ACM (JACM)*, 60(1):1–16, 2013.

Pablo Hernandez-Leal, Bilal Kartal, and Matthew E Taylor. Is multiagent deep reinforcement learning the answer or the question? a brief survey. *Learning*, 21:22, 2018.

Pablo Hernandez-Leal, Bilal Kartal, and Matthew E Taylor. A survey and critique of multiagent deep reinforcement learning. *Autonomous Agents and Multi-Agent Systems*, 33(6):750–797, 2019.

Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A two-timescale framework for bilevel optimization: Complexity analysis and application to actor-critic. *arXiv preprint arXiv:2007.05170*, 2020.

Junling Hu and Michael P Wellman. Nash q-learning for general-sum stochastic games. *Journal of machine learning research*, 4(Nov):1039–1069, 2003.

Natasha Jaques, Asma Ghandeharioun, Judy Hanwen Shen, Craig Ferguson, Agata Lapedriza, Noah Jones, Shixiang Gu, and Rosalind Picard. Way off-policy batch deep reinforcement learning of implicit human preferences in dialog. *arXiv preprint arXiv:1907.00456*, 2019.

Zeyu Jia, Lin F Yang, and Mengdi Wang. Feature-based q-learning for two-player stochastic games. *arXiv preprint arXiv:1906.00423*, 2019.

Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan. Is q-learning provably efficient? *arXiv preprint arXiv:1807.03765*, 2018.

Chi Jin, Akshay Krishnamurthy, Max Simchowitz, and Tiancheng Yu. Reward-free exploration for reinforcement learning. In *International Conference on Machine Learning*, pp. 4870–4879. PMLR, 2020a.

Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, pp. 2137–2143. PMLR, 2020b.

Ying Jin, Zhuoran Yang, and Zhaoran Wang. Is pessimism provably efficient for offline RL? *arXiv preprint arXiv:2012.15085*, 2020c.

Xin Kang and Yongdong Wu. Incentive mechanism design for heterogeneous peer-to-peer networks: A stackelberg game approach. *IEEE Transactions on Mobile Computing*, 14(5):1018–1030, 2014.

Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. Morel: Model-based offline reinforcement learning. *arXiv preprint arXiv:2005.05951*, 2020.

Dmytro Korzhyk, Zhengyu Yin, Christopher Kiekintveld, Vincent Conitzer, and Milind Tambe. Stackelberg vs. Nash in security games: An extended investigation of interchangeability, equivalence, and uniqueness. *Journal of Artificial Intelligence Research*, 41:297–327, 2011.

Aviral Kumar, Justin Fu, George Tucker, and Sergey Levine. Stabilizing off-policy q-learning via bootstrapping error reduction. *arXiv preprint arXiv:1906.00949*, 2019.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *arXiv preprint arXiv:2006.04779*, 2020.

Michail Lagoudakis and Ron Parr. Value function approximation in zero-sum markov games. *arXiv preprint arXiv:1301.0580*, 2012.

Romain Laroche, Paul Trichelair, and Remi Tachet Des Combes. Safe policy improvement with baseline bootstrapping. In *International Conference on Machine Learning*, pp. 3652–3661. PMLR, 2019.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Joshua Letchford, Vincent Conitzer, and Kamesh Munagala. Learning and approximating the optimal strategy to commit to. In *International Symposium on Algorithmic Game Theory*, pp. 250–262. Springer, 2009.

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

Michael L Littman. Friend-or-foe q-learning in general-sum games. In *ICML*, volume 1, pp. 322–328, 2001.

Qiang Liu, Lihong Li, Ziyang Tang, and Dengyong Zhou. Breaking the curse of horizon: Infinite-horizon off-policy estimation. *arXiv preprint arXiv:1810.12429*, 2018.

Qinghua Liu, Tiancheng Yu, Yu Bai, and Chi Jin. A sharp analysis of model-based reinforcement learning with self-play. *arXiv preprint arXiv:2010.01604*, 2020a.

Yao Liu, Adith Swaminathan, Alekh Agarwal, and Emma Brunskill. Provably good batch reinforcement learning without great exploration. *arXiv preprint arXiv:2007.08202*, 2020b.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

Rémi Munos and Csaba Szepesvári. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9(5), 2008.

Ashvin Nair, Murtaza Dalal, Abhishek Gupta, and Sergey Levine. Accelerating online reinforcement learning with offline datasets. *arXiv preprint arXiv:2006.09359*, 2020.

John F Nash. *Non-Cooperative Games*. Princeton University Press, 2016.

Afshin OroojlooyJadid and Davood Hajinezhad. A review of cooperative multi-agent deep reinforcement learning. *arXiv preprint arXiv:1908.03963*, 2019.

Binghui Peng, Weiran Shen, Pingzhong Tang, and Song Zuo. Learning optimal strategies to commit to. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 2149–2156, 2019.

Julien Perolat, Bruno Scherrer, Bilal Piot, and Olivier Pietquin. Approximate dynamic programming for two-player zero-sum markov games. In *International Conference on Machine Learning*, pp. 1321–1329. PMLR, 2015.

Aravind Rajeswaran, Igor Mordatch, and Vikash Kumar. A game theoretic framework for model based reinforcement learning. In *International Conference on Machine Learning*, pp. 7953–7963. PMLR, 2020.

Paria Rashidinejad, Banghua Zhu, Cong Ma, Jiantao Jiao, and Stuart Russell. Bridging offline reinforcement learning and imitation learning: A tale of pessimism. *arXiv preprint arXiv:2103.12021*, 2021.

Lillian J Ratliff and Tanner Fiez. Adaptive incentive design. *IEEE Transactions on Automatic Control*, 2020.

Lillian J Ratliff, Ming Jin, Ioannis C Konstantakopoulos, Costas Spanos, and S Shankar Sastry. Social game for building energy efficiency: Incentive design. In *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1011–1018. IEEE, 2014.

Tim Roughgarden. Stackelberg scheduling strategies. *SIAM journal on computing*, 33(2):332–350, 2004.

Bruno Scherrer, Mohammad Ghavamzadeh, Victor Gabillon, Boris Lesner, and Matthieu Geist. Approximate modified policy iteration and its application to the game of tetris. *J. Mach. Learn. Res.*, 16:1629–1676, 2015.

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

Aaron Sidford, Mengdi Wang, Lin Yang, and Yinyu Ye. Solving discounted stochastic two-player games with near-optimal time and sample complexity. In *International Conference on Artificial Intelligence and Statistics*, pp. 2992–3002. PMLR, 2020.

Noah Y Siegel, Jost Tobias Springenberg, Felix Berkenkamp, Abbas Abdolmaleki, Michael Neunert, Thomas Lampe, Roland Hafner, Nicolas Heess, and Martin Riedmiller. Keep doing what worked: Behavioral modelling priors for offline reinforcement learning. *arXiv preprint arXiv:2002.08396*, 2020.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

Marwaan Simaan and Jose B Cruz. On the stackelberg strategy in nonzero-sum games. *Journal of Optimization Theory and Applications*, 11(5):533–555, 1973.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Richard S Sutton, David A McAllester, Satinder P Singh, Yishay Mansour, et al. Policy gradient methods for reinforcement learning with function approximation. In *NIPs*, volume 99, pp. 1057–1063. Citeseer, 1999.

Milind Tambe. *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge university press, 2011.

Yi Tian, Yuanhao Wang, Tiancheng Yu, and Suvrit Sra. Provably efficient online agnostic learning in markov games. *arXiv preprint arXiv:2010.15020*, 2020.

Bernhard Von Stengel and Shmuel Zamir. Leadership games with convex strategy sets. *Games and Economic Behavior*, 69(2):446–457, 2010.

Ziyu Wang, Alexander Novikov, Konrad Żołna, Jost Tobias Springenberg, Scott Reed, Bobak Shahriari, Noah Siegel, Josh Merel, Caglar Gulcehre, Nicolas Heess, et al. Critic regularized regression. *arXiv preprint arXiv:2006.15134*, 2020.

Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

Chen-Yu Wei, Yi-Te Hong, and Chi-Jen Lu. Online reinforcement learning in stochastic games. *arXiv preprint arXiv:1712.00579*, 2017.

Yifan Wu, George Tucker, and Ofir Nachum. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361*, 2019.

Qiaomin Xie, Yudong Chen, Zhaoran Wang, and Zhuoran Yang. Learning zero-sum simultaneous-move markov games using function approximation and correlated equilibrium. In *Conference on Learning Theory*, pp. 3674–3682. PMLR, 2020.

Tengyang Xie and Nan Jiang. $Q^\star$-approximation schemes for batch reinforcement learning: A theoretical comparison. *arXiv preprint arXiv:2003.03924*, 2020.

Lin Yang and Mengdi Wang. Sample-optimal parametric q-learning using linearly additive features. In *International Conference on Machine Learning*, pp. 6995–7004. PMLR, 2019.

Lin Yang and Mengdi Wang. Reinforcement learning in feature space: Matrix bandit, kernels, and regret bound. In *International Conference on Machine Learning*, pp. 10746–10756. PMLR, 2020.

Zhuoran Yang, Chi Jin, Zhaoran Wang, Mengdi Wang, and Michael I Jordan. Bridging exploration and general function approximation in reinforcement learning: Provably efficient kernel and neural value iterations. *arXiv preprint arXiv:2011.04622*, 2020.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *arXiv preprint arXiv:2005.13239*, 2020.

Andrea Zanette and Emma Brunskill. Tighter problem-dependent regret bounds in reinforcement learning without domain knowledge using value function bounds. In *International Conference on Machine Learning*, pp. 7304–7312. PMLR, 2019.

Andrea Zanette, David Brandfonbrener, Emma Brunskill, Matteo Pirotta, and Alessandro Lazaric. Frequentist regret bounds for randomized least-squares value iteration. In *International Conference on Artificial Intelligence and Statistics*, pp. 1954–1964. PMLR, 2020a.

Andrea Zanette, Alessandro Lazaric, Mykel Kochenderfer, and Emma Brunskill. Learning near optimal policies with low inherent bellman error. In *International Conference on Machine Learning*, pp. 10978–10989. PMLR, 2020b.

Kaiqing Zhang, Zhuoran Yang, and Tamer Başar. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *arXiv preprint arXiv:1911.10635*, 2019.

Kaiqing Zhang, Sham M Kakade, Tamer Başar, and Lin F Yang. Model-based multi-agent rl in zero-sum markov games with near-optimal sample complexity. *arXiv preprint arXiv:2007.07461*, 2020a.

Zihan Zhang, Xiangyang Ji, and Simon S Du. Is reinforcement learning more difficult than bandits? a near-optimal algorithm escaping the curse of horizon. *arXiv preprint arXiv:2009.13503*, 2020b.

Zihan Zhang, Yuan Zhou, and Xiangyang Ji. Almost optimal model-free reinforcement learningvia reference-advantage decomposition. *Advances in Neural Information Processing Systems*, 33, 2020c.

Yulai Zhao, Yuandong Tian, Jason D Lee, and Simon S Du. Provably efficient policy gradient methods for two-player zero-sum markov games. *arXiv preprint arXiv:2102.08903*, 2021.

Stephan Zheng, Alexander Trott, Sunil Srinivasa, Nikhil Naik, Melvin Gruesbeck, David C Parkes, and Richard Socher. The AI economist: Improving equality and productivity with AI-driven tax policies. *arXiv preprint arXiv:2004.13332*, 2020.

Ying-Ping Zheng, Tamer Basar, and Jose B Cruz. Stackelberg strategies and incentives in multiperson deterministic decision problems. *IEEE transactions on Systems, Man, and Cybernetics*, (1): 10–24, 1984.

Dongruo Zhou, Quanquan Gu, and Csaba Szepesvari. Nearly minimax optimal reinforcement learning for linear mixture markov decision processes. *arXiv preprint arXiv:2012.08507*, 2020.