Sample-Efficient Learning of Stackelberg Equilibria in General-Sum Games

Anonymous Author(s)
Affiliation
Address
email

Abstract

Real world applications such as economics and policy making often involve solving multi-agent games with two unique features: (1) The agents are inherently *asymmetric* and partitioned into leaders and followers; (2) The agents have different reward functions, thus the game is *general-sum*. The majority of existing results in this field focuses on either symmetric solution concepts (e.g. Nash equilibrium) or zero-sum games. It remains vastly open how to learn the *Stackelberg equilibrium*—an asymmetric analog of the Nash equilibrium—in general-sum games efficiently from samples. This paper initiates the theoretical study of sample-efficient learning of the Stackelberg equilibrium, in the bandit feedback setting where we only observe noisy samples of the reward. We consider three representative two-player general-sum games: bandit games, bandit-reinforcement learning (bandit-RL) games, and linear bandit games. In all these games, we identify a fundamental gap between the exact value of the Stackelberg equilibrium and its estimated version using finitely many noisy samples, which can not be closed information-theoretically regardless of the algorithm. We then establish sharp positive results on sample-efficient learning of Stackelberg equilibrium with value optimal up to the gap identified above, with matching lower bounds in the dependency on the gap, error tolerance, and the size of the action spaces. Overall, our results unveil unique challenges in learning Stackelberg equilibria under noisy bandit feedback, which we hope could shed light on future research on this topic.

1 Introduction

Real-world problems such as economic design and policy making can often be modeled as multi-agent games that involves two levels of thinking: The policy maker—as a player in this game—needs to reason about the other player's optimal behaviors given her decision, in order to inform her own optimal decision making. Consider for example the optimal taxation problem in the AI Economist [48], a game modeling a real-world social-economic system involving a *leader* (e.g. the government) and a group of interacting *followers* (e.g. citizens). The leader sets a tax rate which determines an economics-like game for the followers; the followers then play in this game with the objective to maximize their own reward (such as individual productivity). However, the goal of the leader is to maximize her own reward (such as overall equality) which is in general different from the followers' rewards, making these games *general-sum* [37]. Such two-level thinking appears broadly in other applications as well such as in automated mechanism design [11, 12], optimal auctions [10, 14], security games [41], reward shaping [24], and so on.

Another key feature in such games is that the players are *asymmetric*, and they act in turns: the leader first plays, then the follower sees the leader's action and then adapts to it. This makes symmetric solution concepts such as Nash equilibrium [29] not always appropriate. A more natural solution concept for these games is the *Stackelberg equilibrium*: the leader's optimal strategy, assuming the

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

followers play their best response to the leader [40, 13]. The Stackelberg equlibrium is often the desired solution concept in many of the aforementioned applications. Furthermore, it is of compelling interest to understand the learning of Stackelberg equilibria from *samples*, as it is often the case that we can only learn about the game through interactively deploying policies and observing the (noisy) feedback from the game [48].

Despite the motivations, theoretical studies of learning Stackelberg equilibria in general-sum games remain vastly open, in particular when we can only learn from random samples. A line of work provides guarantees for finding Stackelberg equilibria in general-sum games, but restricts attention to either the full observation setting (so that the exact game is observable) or with an exact best-response oracle [13, 26, 44, 33]. These results lay out a foundation for analyzing the Stackelberg equilibrium, but do not generalize to the bandit feedback setting in which the game can only be learned from random samples. Another line of work considers the sample complexity of learning the Nash equilibrium in Markov games [34, 4, 5, 27, 46, 47], which also do not imply algorithms for finding the Stackelberg equilibrium in these games as the Nash is in general different from the Stackelberg equilibrium in general-sum games. So far, it is unclear how to learn Stackelberg in general-sum games sample-efficiently.

In this work, we study the sample complexity of learning Stackelberg equilibrium in general-sum games. We focus on general-sum games with two players (one leader and one follower), in which we wish to learn an approximate Stackelberg equilbrium for the leader from random samples. Our contributions can be summarized as follows.

- As a warm-up, we consider *bandit games* in which the two players play an action in turns and observe their own rewards. We identify a fundamental gap between the exact Stackelberg value and its estimated version from finite samples, which cannot be closed information-theoretically regardless of the algorithm (Section 3.1). We then propose a rigorous definition gap_ε for this gap, and show that it is possible to sample-efficiently learn the $(\mathsf{gap}_\varepsilon + \varepsilon)$-approximate Stackelberg equilibrium with $\widetilde{O}(AB/\varepsilon^2)$ samples, where A, B are the number of actions for the two players. We further show a matching lower bound $\Omega(AB/\varepsilon^2)$ (Section 3.2). We also establish similar results for learning Stackelberg in simultaneous matrix games (Appendix B).
- We consider *bandit-RL games* in which the leader's action determines an episodic Markov Decision Process (MDP) for the follower. We show that a $(\mathsf{gap}_\varepsilon + \varepsilon)$ approximate Stackelberg equilibrium for the leader can be found in $\widetilde{O}(H^5 S^2 AB/\varepsilon^2)$ episodes of play, where H, S are the horizon length and number of states for the follower's MDP, and A, B are the number of actions for the two players (Section 4). Our algorithm utilizes recently developed reward-free reinforcement learning techniques to enable fast exploration for the follower within the MDPs.
- Finally, we consider *linear bandit games* in which the action spaces for the two players can be arbitrarily large, but the reward is a linear function of a d-dimensional feature representation of the actions. We design an algorithm that achieves $\widetilde{O}(d^2/\varepsilon^2)$ sample complexity upper bound for linear bandit games (Section 5). This only depends polynomially on the feature dimension instead of the size of the action spaces, and has at most an $\widetilde{O}(d)$ gap from the lower bound.

1.1 Related work

Since the seminal paper of [43], notions of equilibria in games and their algorithmic computation have received wide attention [see, e.g., 9, 39]. For the scope of this paper, this section focuses on reviewing results that related to learning Stackelberg equilibria.

Learning Stackelberg equilibria in zero-sum games The first category of results study two-player zero-sum games, where the rewards of the two players sum to zero. Most results along this line focus on the bilinear or convex-concave setting [see, e.g., 20, 31, 30, 36, 16], where the Stackelberg equilibrium coincide with the Nash equilibrium due to Von Neumann's minimax theorem [43]. Results for learning Stackelberg equilibria beyond convex-concave setting are much more recent, with Rafique et al. [35], Nouiehed et al. [32] considering the nonconvex-concave setting, and Fiez et al. [15], Jin et al. [19], Marchesi et al. [28] considering the nonconvex-nonconcave setting. Marchesi et al. [28] provide sample complexity results for learning Stackelberg with infinite strategy spaces, using discretization techniques that may scale exponentially in the dimension without further assumptions on the problem structure.

We remark that a crucial property of zero-sum games is that any two strategies giving similar rewards for the follower will also give similar rewards for the leader. This is no longer true in general-sum games, and prevents most statistical results for learning Stackelberg equilibria in the zero-sum setting from generalizing to the general-sum setting.

Learning Stackelberg equilibria in general-sum games The computational complexity of finding Stackelberg equilibria in games with simultaneous play ("computing optimal strategy to commit to") is studied in [13, 25, 44, 21, 1, 7]. These results assume full observation of the payoff function, and show that several versions of matrix games and extensive-form (multi-step) games admit polynomial time algorithms. Another line of work considers learning Stackelberg with a "best response oracle" [26, 6, 33], which is in general incomparable with our bandit feedback oracle (their oracle allows querying any leader strategy and returns the *exact* best response for that strategy) and do not imply sample complexity results in our setting. Fiez et al. [15] study the local convergence of first-order algorithms for finding Stackelberg equilibria in general-sum games. Their result also assumes exact feedback and do not allow sampling errors. Lastly, the AI Economist [48] studied the optimal taxation problem by learning the Stackelberg equilibrium via a two-level reinforcement learning approach.

Learning equilibria in Markov games A recent line of results [4, 5, 46, 47] consider learning Markov games [38]—a generalization of Markov decision process to the multi-agent setting. We remark that all three settings studied in this paper can be cast as special cases of general-sum Markov games, which is studied by [27]. In particular, Liu et al. [27] provides sample complexity guarantees for finding Nash equilibria, correlated equilibria, or coarse correlated equilibria of the general-sum Markov games. These Nash-finding algorithms are related to our setting, but do not imply results for learning Stackelberg (see Section 3.2 and Appendix C.5 for detailed discussions).

2 Preliminaries

Bandit games A general-sum two-player bandit game can be described by a tuple $M = (\mathcal{A}, \mathcal{B}, r_1, r_2)$, which defines the following game played by two players, a *leader* and a *follower*:

- The leader plays an action $a \in \mathcal{A}$, with $|\mathcal{A}| = A$.
- The follower sees the action played by the leader, and plays an action $b \in \mathcal{B}$, with $|\mathcal{B}| = B$.
- The follower observes a (potentially random) reward $r_2(a, b) \in [0, 1]$. The leader also observes her own reward $r_1(a, b) \in [0, 1]$.

Note that this is a special case of a general-sum turn-based Markov game with two steps [38, 4]. This game is also a turn-based variant of the simultaneous matrix game considered in [13, 26] (for which we also provide results in Appendix B).

Best response, Stackelberg equilibrium Let $\mu_i(a, b) := \mathbb{E}[r_i(a, b)]$ $(i = 1, 2)$ denote the mean rewards. For each leader action a, the *best response set* $\mathsf{BR}_0(a)$ is the set of follower actions that maximize $\mu_2(a, \cdot)$:

$$\mathsf{BR}_0(a) := \left\{ b : \mu_2(a, b) = \max_{b' \in \mathcal{B}} \mu_2(a, b') \right\}. \tag{1}$$

Given the best-response set $\mathsf{BR}_0(a)$, we define the function $\phi_0 : \mathcal{A} \to [0, 1]$ as the leader's value function when the follower plays the worst-case best response (henceforth the"exact ϕ-function"):

$$\phi_0(a) := \min_{b \in \mathsf{BR}_0(a)} \mu_1(a, b), \tag{2}$$

This is the value function for the leader action a, assuming the follower plays the best response to a and breaks ties in the best response set against the leader's favor. This is known as *pessimistic tie breaking* and provides a worst-case guarantee for the leader [13]. We remark that here restricting b to pure strategies (deterministic actions) is without loss of generality, as there is at least one pure strategy that maximizes $\mu_2(a, \cdot)$ and (among the maximizers) minimize $\mu_1(a, \cdot)$, among all mixed strategies.

The Stackelberg Equilibrium (henceforth also "Stackelberg") for the leader is the "best response to the best response", i.e. any action $a_\star$ that maximizes ϕ_0 [40]:

$$a_\star \in \arg\max_{a \in \mathcal{A}} \phi_0(a). \tag{3}$$

We are interested in finding approximate solutions to the Stackelberg equilibrium, that is, an action $\widehat{a}$ that approximately maximizes $\phi_0(a)$. Note that as the leader's action is seen by the follower, it suffices to only consider pure strategies for the leader too (this is equivalent to the "optimal committment to pure strategies" problem of [13]). We also remark that, while we consider pessimistic tie breaking (definition (2)) in this paper, similar results hold in the optimistic setting as well in which the follower breaks ties in favor of the leader. We defer the statements and proofs of these results to Appendix A.

Real-world example (AI Economist): Consider the following (simplified) optimal taxation problem in the AI Economist [48] as an example of a bandit game. The government (leader) determines the tax rate $a \in \mathcal{A}$ for the follower. The citizen (follower) then chooses the amount of labor $b \in \mathcal{B}$ she wishes to perform. The rewards for the two players are different in general: For example, the citizen's reward $r_2(a, b)$ can be her post-tax income per labor, and the government's reward $r_1(a, b)$ can be a weighted average of its tax income and some measure of the citizen's welfare (e.g. not too much labor). We remark that the actual AI Economist is more similar to a *bandit-RL game* where the follower plays sequentially in an MDP determined by the leader, which we study in Section 4.

Sample-efficient learning with bandit feedback In this paper we consider the *bandit feedback* setting, that is, the algorithm cannot directly observe the mean rewards $\mu_1(\cdot, \cdot)$ and $\mu_2(\cdot, \cdot)$, and can only query (a, b) and obtain random samples $(r_1(a, b), r_2(a, b))$. Our goal is to determine the number of samples in order to find an approximate maximizer of $\phi_0(a)$.

Note that the bandit feedback setting assumes observation noise in the rewards. As we will see in Section 3, this noise turns out to bring in a fundamental challenge for learning Stackelberg equilibria that is not present in (and thus not directly solved by) existing work on learning Stackelberg, which assumes either exact observation of the mean rewards [13, 25], or the best-response oracle that can query a and obtain the exact best response set $\mathsf{BR}_0(a)$ [26, 33].

Markov decision processes We also present the basics of a Markov Decision Processes (MDPs), which will be useful for the bandit-RL games in Section 4. We consider episodic MDPs defined by a tuple $(H, \mathcal{S}, \mathcal{B}, d^1, \mathbb{P}, r)$, where H is the horizon length, $\mathcal{S}$ is the state space, $\mathcal{B}$ is the action space[1], $\mathbb{P} = \{\mathbb{P}_h(\cdot|s, b) : h \in [H], s \in \mathcal{S}, b \in \mathcal{B}\}$ is the transition probabilities, and $r = \{r_h : \mathcal{S} \times \mathcal{B} \to [0, 1], h \in [H]\}$ are the (potentially random) reward functions. A policy $\pi = \left\{\pi_h^b(\cdot|s) \in \Delta_{\mathcal{B}} : h \in [H], s \in \mathcal{S}\right\}$ for the player is a set of probability distributions over actions given the state.

In this paper we consider the exploration setting as the protocol of interacting with MDPs, similar as in [3, 17]. The learning agent is able to play episodes repeatedly, where in each episode at step $h \in \{1, \ldots, H\}$, the agent observes state $s_h \in \mathcal{S}$, takes an action $b_h \sim \pi_h(\cdot|s_h)$, observes her reward $r_h = r_h(s_h, b_h) \in [0, 1]$, and transits to the next state $s_{h+1} \sim \mathbb{P}_h(\cdot|s_h, b_h)$. The initial state is received from the MDP: $s_1 \sim d^1(\cdot)$. The overall value function (return) of a policy π is defined as $V(\pi) := \mathbb{E}_\pi\left[\sum_{h=1}^H r_h(s_h, b_h)\right]$.

3 Warm-up: bandit games

3.1 Hardness of maximizing ϕ_0 from samples

Given the exact ϕ-function ϕ_0 (2), a natural notion of approximate Stackelberg equilibrium is to find an action $\widehat{a}$ that is ε near-optimal for maximizing ϕ_0:

$$\phi_0(\widehat{a}) \geq \max_{a \in \mathcal{A}} \phi_0(a) - \varepsilon. \tag{4}$$

However, the following lower bound shows that, in the worst case, it is hard to find such $\widehat{a}$ from finite samples.

[1] The notation $\mathcal{B}$ indicates that the MDP is played by the follower (cf. Section 4); we reserve $\mathcal{A}$ as the leader's action space in this paper.

Theorem 1 ($\Omega(1)$ lower bound for maximizing ϕ_0)**.** *For any sample size n and any algorithm for maximizing ϕ_0 that outputs an action $\widehat{a} \in \mathcal{A}$, there exists a bandit game with $A = B = 2$ on which the algorithm must suffer from $\Omega(1)$ error with probability at least $1/3$:*

$$\phi_0(\widehat{a}) \leq \max_{a \in \mathcal{A}} \phi_0(a) - 1/2.$$

Theorem 1 shows that, no matter how large the sample size n is, any algorithm in the worst-case have to suffer from an $\Omega(1)$ lower bound for maximizing ϕ_0 (i.e. determining the Stackelberg equilibrium for the leader). This result stems from a *hardness of determining the best response* $\mathsf{BR}_0(a)$ *exactly* from samples. (See Table 2 for the construction of the hard instance and Appendix C.1 for the full proof of Theorem 1.) This is in stark contrast to the standard $1/\sqrt{n}$ type learning result in finding other solution concepts such as the Nash equilibrium [4, 27], and suggests a new fundamental challenge to learning Stackelberg equilibrium from samples.

3.2 Learning Stackelberg with value optimal up to gap

The lower bound in Theorem 1 shows that approximately maximizing ϕ_0 is information-theoretically hard. Motivated by this, we consider in turn a slightly relaxed notion of optimality, in which we consider maximizing ϕ_0 only up to the *gap* between ϕ_0 and its counterpart using ε-approximate best responses. More concretely, define the ε-approximate versions of the best response set and ϕ-function as

$$\phi_\varepsilon(a) := \min_{b \in \mathsf{BR}_\varepsilon(a)} \mu_1(a, b),$$
$$\mathsf{BR}_\varepsilon(a) := \left\{ b \in \mathcal{B} : \mu_2(a, b) \geq \max_{b'} \mu_2(a, b') - \varepsilon \right\}.$$

These definitions are similar to the vanilla BR_0 and ϕ_0 in (1) and (2), except that we allow any ε-approximate best response to be considered as a valid response to the leader action. Observe we always have $\mathsf{BR}_\varepsilon(a) \supseteq \mathsf{BR}_0(a)$ and $\phi_\varepsilon(a) \leq \phi_0(a)$. We then define the *gap* of the game for any $\varepsilon \in (0, 1)$ as

$$\mathsf{gap}_\varepsilon := \max_{a \in \mathcal{A}} \phi_0(a) - \max_{a \in \mathcal{A}} \phi_\varepsilon(a) \geq 0. \tag{5}$$

This gap_ε is discontinuous in ε in general and can be as large as $\Theta(1)$ for any $\varepsilon > 0$. See Figure 1 for an illustration for a typical $\max_{a \in \mathcal{A}} \phi_\varepsilon(a)$ against ε.

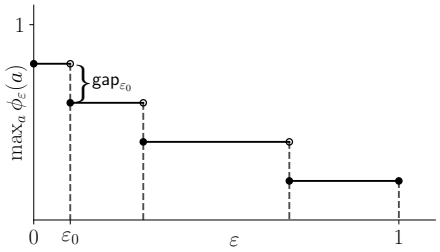

Figure 1: Illustration of $\max_{a \in \mathcal{A}} \phi_\varepsilon(a)$ and gap_ε as a function of ε. The quantity $\mathsf{gap}_{\varepsilon_0}$ can be $\Omega(1)$ for arbitrarily small ε_0.

With the definition of the gap, we are now ready to state our main result, which shows that it is possible to sample-efficiently learn Stackelberg Equilibria with value up to $(\mathsf{gap}_\varepsilon + \varepsilon)$. The proof can be found in Appendix C.3.

Theorem 2 (Learning Stackelberg in bandit games)**.** *For any bandit game and $\varepsilon \in (0, 1)$, Algorithm 1 outputs $(\widehat{a}, \widehat{b})$ such that with probability at least $1 - \delta$,*

$$\phi_0(\widehat{a}) \geq \phi_{\varepsilon/2}(\widehat{a}) \geq \max_{a \in \mathcal{A}} \phi_0(a) - \mathsf{gap}_\varepsilon - \varepsilon,$$
$$\mu_2(\widehat{a}, \widehat{b}) \geq \max_{b' \in \mathcal{B}} \mu_2(\widehat{a}, b') - \varepsilon$$

with $n = \widetilde{O}\big(AB/\varepsilon^2\big)$ samples, where $\widetilde{O}(\cdot)$ hides log factors. Further, the algorithm runs in $O(n) = \widetilde{O}(AB/\varepsilon^2)$ time.

Implications; Overview of algorithm Theorem 2 shows that it is possible to learn $\widehat{a}$ that maximizes $\phi_0(a)$ up to $(\mathsf{gap}_\varepsilon + \varepsilon)$ accuracy, using $\widetilde{O}(AB/\varepsilon^2)$ samples. The quantity gap_ε is not bounded and can be as large as $\Theta(1)$ for any ε (see Lemma C.1 for a formal statement); however the gap is non-increasing as we decrease ε. In the situation where for every a the best follower action for $\mu_2(a, \cdot)$ is at least ε_0-better than the second best action, then for $\varepsilon < \varepsilon_0$ we have $\mathsf{gap}_\varepsilon = 0$ and Theorem 2

Algorithm 1 Learning Stackelberg in bandit games

Require: Target accuracy $\varepsilon > 0$.

set $N \leftarrow C\log(AB/\delta)/\varepsilon^2$ for some constant $C > 0$.

1: Query each $(a, b) \in \mathcal{A} \times \mathcal{B}$ for N times and obtain $\{r_1^{(j)}(a, b), r_2^{(j)}(a, b)\}_{j=1}^N$.

2: Construct empirical estimates of the means $\widehat{\mu}_i(a, b) = \frac{1}{N}\sum_{j=1}^N r_i^{(j)}(a, b)$ for $i = 1, 2$.

3: Construct approximate best response sets and values for all $a \in \mathcal{A}$:

$$\widehat{\phi}_{3\varepsilon/4}(a) := \min_{b \in \widehat{\mathsf{BR}}_{3\varepsilon/4}(a)} \widehat{\mu}_1(a, b), \quad \text{where} \quad \widehat{\mathsf{BR}}_{3\varepsilon/4}(a) := \left\{ b : \widehat{\mu}_2(a, b) \geq \max_{b' \in \mathcal{B}} \widehat{\mu}_2(a, b') - 3\varepsilon/4 \right\}.$$

4: Output $(\widehat{a}, \widehat{b})$, where $\widehat{a} = \arg\max_{a \in \mathcal{A}} \widehat{\phi}_{3\varepsilon/4}(a)$, $\widehat{b} = \arg\min_{b \in \widehat{\mathsf{BR}}_{3\varepsilon/4}(\widehat{a})} \widehat{\mu}_1(\widehat{a}, b)$.

implies an ε-optimal Stackelberg guarantee. In general, Theorem 2 presents a "best-effort" positive result for learning Stackelberg under this relaxed notion of optimality. To the best of our knowledge, this is the first result for sample-efficient learning of Stackelberg equilibrium in general-sum games with bandit feedbacks. We remark that Theorem 2 also provides a near-optimality guarantee for $\phi_{\varepsilon/2}(\widehat{a})$ which is slightly stronger than ϕ_0 (since $\phi_{\varepsilon/2}(\widehat{a}) \leq \phi_0(\widehat{a})$), and guarantees the learned $\widehat{b}$ is indeed an ε-approximate best response of $\widehat{a}$.

The main step in Algorithm 1 is to construct approximate best response sets $\widehat{\mathsf{BR}}_{3\varepsilon/4}(a)$ for all $a \in \mathcal{A}$ based on the empirical estimates of the rewards. Through concentration, we argue that $\widehat{\mathsf{BR}}_{3\varepsilon/4}(a)$ is a good approximation of the true best response sets in the sense that $\mathsf{BR}_{\varepsilon/2}(a) \subseteq \widehat{\mathsf{BR}}_{3\varepsilon/4}(a) \subseteq \mathsf{BR}_{\varepsilon}(a)$ holds for all $a \in \mathcal{A}$, from which the Stackelberg guarantee follows.

Irreducibility to Nash-finding algorithms We also remark that our bandit game is equivalent to a turn-based general-sum Markov game with two steps, A states, and (A, B) actions [4]. Further, the Stackelberg equilibrium $a_\star$ along with a follower policy that plays the best response (with pessimistic tie-breaking) constitutes a Nash equilibrium for that Markov game (see Appendix C.5 for a formal statement and proof). However, existing Nash-finding algorithms for general-sum Markov games such as `Multi-Nash-VI` [27] *do not* imply an algorithm for finding the Stackelberg equilibrium. This is because general-sum games have multiple Nash equilibria (with different values) in general [37], and these existing Nash-finding algorithms cannot pre-specify which Nash to output.

Lower bound We accompany Theorem 2 by an $\Omega(AB/\varepsilon^2)$ sample complexity lower bound, showing that Theorem 2 achieves the optimal sample complexity up to logarithmic factors.

Theorem 3 (Lower bound for bandit games)**.** *There exists an absolute constant $c > 0$ such that the following holds. For any $\varepsilon \in (0, c)$, $g \in [0, c)$, any $A, B \geq 3$, and any algorithm that queries $N \leq c\left[AB/\varepsilon^2\right]$ samples and outputs an estimate $\widehat{a} \in \mathcal{A}$, there exists a bandit game M on which $\mathsf{gap}_\varepsilon = g$ and the algorithm suffers from $(g + \varepsilon)$ error:*

$$\phi_{\varepsilon/2}(\widehat{a}) \leq \phi_0(\widehat{a}) \leq \max_{a \in \mathcal{A}} \phi_0(a) - g - \varepsilon$$

with probability at least $1/3$.

This lower bound shows the tightness of Theorem 2, and suggests that $(\mathsf{gap}_\varepsilon + \varepsilon)$ suboptimality is perhaps a sensible learning goal, as for any algorithm and any value of $g \geq 0$ there exists a game with $\mathsf{gap}_\varepsilon = g$, on which the algorithm has to suffer from $(g + \varepsilon)$ error, if the number of samples is at most $O(AB/\varepsilon^2)$. The proof of Theorem 3 is deferred to Appendix C.4.

4 Bandit-RL games

In this section, we investigate learning Stackelberg equilibrium in bandit-RL games, in which each leader's action determines an episodic Markov Decision Process (MDP) for the follower. This setting extends the two-player bandit games by allowing the follower to play sequentially, and has strong practical motivations in particular in policy making problems involving sequential plays for the follower, such as the optimal taxation problem in the AI Economist [48].

Setting A bandit-RL game is described by the leader's action set $\mathcal{A}$ (with $|\mathcal{A}| = A$), and a family of MDPs $M = \{M^a : a \in \mathcal{A}\}$. Each leader action $a \in \mathcal{A}$ determines an episodic MDP $M^a = (H, \mathcal{S}, \mathcal{B}, \mathbb{P}^a, r_{1,h}(a, \cdot, \cdot), r_{2,h}(a, \cdot, \cdot))$ that contains H steps, S states, B actions, with two reward functions r_1 and r_2. In each episode of play,

- The leader plays action $a \in \mathcal{A}$.
- The follower sees this action and enters the MDP M^a. She observes the deterministic[2] initial state s_1, and plays in M^a with exploration feedback for one episode.
- While the follower plays in the MDP, she observes reward $r_{2,h}(a, s_h, b_h)$, whereas the leader also observes her own reward $r_{1,h}(a, s_h, b_h)$.

We let π^b denote a policy for the follower, and let $V_1(a, \pi^b)$ and $V_2(a, \pi^b)$ denote its value functions (in M^a) for the leader and the follower respectively.

Similar as in bandit games, we define the ε-approximate best-response set $\mathsf{BR}_\varepsilon(a)$ and the ε-approximate ϕ-function $\phi_\varepsilon(a)$ for all $\varepsilon \geq 0$ as

$$\phi_\varepsilon(a) := \min_{\pi^b \in \mathsf{BR}_\varepsilon(a)} V_1(a, \pi^b), \quad \text{where} \quad \mathsf{BR}_\varepsilon(a) := \left\{ \pi^b : V_2(a, \pi^b) \geq \max_{\widetilde{\pi}^b} V_2(a, \widetilde{\pi}^b) - \varepsilon \right\}.$$

Define $\mathsf{gap}_\varepsilon = \max_{a \in \mathcal{A}} \phi_0(a) - \max_{a \in \mathcal{A}} \phi_\varepsilon(a)$ similarly as in (5). We are interested in the number of episodes in order to find a $(\mathsf{gap}_\varepsilon + \varepsilon)$ near-optimal Stackelberg equilibrium.

4.1 Algorithm description

At a high level, our algorithm for bandit-RL games is similar as for bandit games – query each leader action $a \in \mathcal{A}$ sufficiently many times, let the follower learn the best response (i.e. best policy for the MDP M^a) for each $a \in \mathcal{A}$, and then choose the leader action that maximizes the best response value function. This requires solving

$$\begin{aligned} \arg\max_{a \in \mathcal{A}} \phi_{3\varepsilon/4}(a) &:= \arg\max_{a \in \mathcal{A}} \min_{\pi^b \in \widehat{\mathsf{BR}}_{3\varepsilon/4}(a)} \widehat{V}_1(a, \pi^b), \\ \widehat{\mathsf{BR}}_{3\varepsilon/4}(a) &:= \left\{ \pi^b : \widehat{V}_2(a, \pi^b) \geq \max_{\widetilde{\pi}^b} \widehat{V}_2(a, \widetilde{\pi}^b) - 3\varepsilon/4 \right\}, \end{aligned} \tag{6}$$

where $\widehat{V}_1$ and $\widehat{V}_2$ are empirical estimates of the true value functions.

Two technical challenges emerge as we instantiate (6). First, the follower not only needs to find her own best policy during the exploration phase, but also has to accurately estimate both her own and the leader's reward over the entire approximate best response set $\widehat{\mathsf{BR}}_{3\varepsilon/4}$ so as to make sure the estimates $\widehat{V}_i(a, \pi^b)$ $(i = 1, 2)$ reliable. Standard fast PAC-exploration algorithms such as those in [3, 17] do not provide such guarantees. We resolve this by applying the *reward-free learning algorithm* of Jin et al. [18] for the follower to explore the environment efficiently while providing value concentration guarantees for multiple rewards and policies. We remark that we slightly generalize the guarantees of [18] to the situation where the rewards are random and have to be estimated from samples.

Second, the problem $\min_{\pi^b \in \widehat{\mathsf{BR}}_{3\varepsilon/4}(a)} \widehat{V}_1(a, \pi^b)$ in (6) requires minimizing a value function over the near-optimal policy set of another value function. We build on the linear programming reformulation in the constrained MDP literature [2] to translate (6) into a linear program `WorstCaseBestResponse`, which adopts efficient solution in $\mathrm{poly}(HSB)$ time [8]. (the description of this subroutine can be found in Algorithm 8 in Appendix D.1). Our full algorithm is described in Algorithm 2.

4.2 Main result

We now state our theoretical guarantee for Algorithm 2. The proof can be found in Appendix D.2.

[2]The general case where s_1 is stochastic reduces to the deterministic case by adding a step $h = 0$ with a single deterministic initial state s_0, which only increases the horizon of the game by 1.

Algorithm 2 Learning Stackelberg in bandit-RL games

Require: Target accuracy $\varepsilon > 0$.

1: **for** $a \in \mathcal{A}$ **do**

2: Let the leader pull arm $a \in \mathcal{A}$ and the follower run the `Reward-Free RL-Explore` algorithm for $N \leftarrow \widetilde{O}(H^5S^2B/\varepsilon^2 + H^7S^4B/\varepsilon)$ episodes, and obtain model estimate $\widehat{M}^a$.

3: Let $(\widehat{V}_1(a,\cdot), \widehat{V}_2(a,\cdot))$ denote the value functions for the model $\widehat{M}^a$.

4: Compute the empirical best response value $\widehat{V}_2^\star(a) := \max_{\pi^b} \widehat{V}_2(a, \pi^b)$ by any optimal planning algorithm (e.g. value iteration) on the empirical MDP $\widehat{M}^a$.

5: Solve the following program

$$\text{minimize}_{\pi^b} \ \ \widehat{V}_1(a, \pi^b) \ \ \text{s.t.} \ \ \pi^b \in \widehat{\mathsf{BR}}_{3\varepsilon/4}(a) := \left\{\pi^b : \widehat{V}_2(a, \pi^b) \geq \widehat{V}_2^\star(a) - 3\varepsilon/4\right\} \tag{7}$$

by subroutine $(\widehat{\pi}^{b,(a)}, \widehat{\phi}_{3\varepsilon/4}(a)) \leftarrow$ `WorstCaseBestResponse`$(\widehat{M}^a, \widehat{V}_2^\star(a) - 3\varepsilon/4)$.

output $(\widehat{a}, \widehat{\pi}^b)$ where $\widehat{a} \leftarrow \arg\max_{a\in\mathcal{A}} \widehat{\phi}_{3\varepsilon/4}(a)$ and $\widehat{\pi}^b \leftarrow \widehat{\pi}^{b,(\widehat{a})}$.

Theorem 4 (Learning Stackelberg in bandit-RL games)**.** *For any bandit-RL game and sufficiently small $\varepsilon \leq O(1/H^2S^2)$, Algorithm 2 with $n = \widetilde{O}(H^5S^2AB/\varepsilon^2 + H^7S^4AB/\varepsilon)$ episodes of play can return $(\widehat{a}, \widehat{\pi}^b)$ such that with probability at least $1-\delta$,*

$$\phi_0(\widehat{a}) \geq \phi_{\varepsilon/2}(\widehat{a}) \geq \max_{a\in\mathcal{A}} \phi_0(a) - \mathsf{gap}_\varepsilon - \varepsilon,$$
$$V_2(\widehat{a}, \widehat{\pi}^b) \geq \max_{\widetilde{\pi}^b} V_2(\widehat{a}, \widetilde{\pi}^b) - \varepsilon,$$

where $\widetilde{O}(\cdot)$ hides $\log(HSAB/\delta\varepsilon)$ factors. Further, the algorithm runs in $\text{poly}(HSAB/\delta\varepsilon)$ time.

Sample complexity, relationship with reward-free RL Theorem 4 shows that for bandit-RL games, the approximate Stackelberg Equilibrium (with value optimal up to $\mathsf{gap}_\varepsilon + \varepsilon$) can be efficiently found with polynomial sample complexity and runtime. In particular, (for small ε) the leading term in the sample complexity scales as $\widetilde{O}(H^5S^2AB/\varepsilon^2)$. Since bandit-RL games include bandit games as a special case, the $\Omega(AB/\varepsilon^2)$ lower bound for bandit games (Theorem 3) apply here and implies that the A, B dependence in Theorem 4 is tight, while the H dependence may be slightly suboptimal.

We also remark the learning goal for the follower in our bandit-RL game is a new RL setting in between the single-reward setting and the full reward-free setting, for which the optimal S dependence is currently unclear. In the single-reward setting, existing fast exploration algorithms such as `UCBVI` [3] only require linear in S episodes for finding a near-optimal policy. In contrast, in the full reward-free setting (follower wants accurate estimation of any reward), it is known $\Omega(S^2)$ is unavoidable [18]. Our setting poses a unique challenge in between: The follower wishes to accurately estimate both r_1, r_2 on all near-optimal policies for r_2. This further renders recent linear in S algorithms for reward-free learning with finitely many rewards [27] not applicable here, as they only guarantee accurate estimation of each reward on near-optimal policies for *that* reward. We believe the optimal sample complexity for bandit-RL games is an interesting open question.

5 Linear bandit games

Setting We consider a bandit game with action space $\mathcal{A}, \mathcal{B}$ that are finite but potentially arbitrarily large, and assume in addition that the reward functions has a linear form

$$r_1(a,b) = \phi(a,b)^\top \theta_1^\star + z_1, \quad r_2(a,b) = \phi(a,b)^\top \theta_2^\star + z_2, \tag{8}$$

where $\phi : \mathcal{A} \times \mathcal{B} \to \mathbb{R}^d$ is a d-dimensional feature map, $\theta_1^\star, \theta_2^\star \in \mathbb{R}^d$ are unknown ground truth parameters for the rewards, and z_1, z_2 are random noise which we assume are mean-zero and 1-sub-Gaussian. Let $\Phi := \{\phi(a,b) : (a,b) \in \mathcal{A} \times \mathcal{B}\}$ denote the set of all possible features. For linear bandit games, we define gap_ε same as definition (5) for bandit games.

Algorithm and guarantee We present our algorithm for linear bandit games in Algorithm 3. Compared with our Algorithm 1 for bandit games, Algorithm 3 takes advantage of the linear structure

Algorithm 3 Learning Stackelberg in linear bandit games

Require: Target accuracy $\varepsilon > 0$.

1: Find $(\mathcal{K}, \rho) \leftarrow \texttt{CoreSet}(\Phi)$ (cf. (10)). Let $\mathcal{K} = \{(a_j, b_j) : 1 \le j \le K\}$ where $K = |\mathcal{K}|$.

2: Query each (a_j, b_j) for $N = O(d \log(d/\delta)/\varepsilon^2)$ times. Let $(\widehat{\mu}_{1,j}, \widehat{\mu}_{2,j})$ denote the empirical mean of the observed rewards over the N queries.

3: Estimate $(\theta_1^\star, \theta_2^\star)$ via weighted least squares

$$\widehat{\theta}_i := \arg\min_{\theta \in \mathbb{R}^d} \sum_{i=1}^{K} \rho(a_j, b_j)\big(\phi(a_j, b_j)^\top \theta_i - \widehat{\mu}_{i,j}\big)^2, \quad i = 1, 2. \tag{9}$$

4: Construct approximate best response sets and values for all $a \in \mathcal{A}$:

$$\widehat{\mathsf{BR}}_{3\varepsilon/4}(a) := \left\{ b : \phi(a, b)^\top \widehat{\theta}_2 \ge \max_{b' \in \mathcal{B}} \phi(a, b')^\top \widehat{\theta}_2 - 3\varepsilon/4 \right\},$$

$$\widehat{\phi}_{3\varepsilon/4}(a) := \min_{b \in \widehat{\mathsf{BR}}_{3\varepsilon/4}(a)} \phi(a, b)^\top \widehat{\theta}_1.$$

5: Output $(\widehat{a}, \widehat{b})$, where $\widehat{a} = \arg\max_{a \in \mathcal{A}} \widehat{\phi}_{3\varepsilon/4}(a)$, $\widehat{b} = \arg\min_{b \in \widehat{\mathsf{BR}}_{3\varepsilon/4}(\widehat{a})} \phi(\widehat{a}, b)^\top \widehat{\theta}_1$.

through the following important modifications: (1) Rather than querying every action pair, we only query (a, b) in a *core set* $\mathcal{K}$ found through the following subroutine

$$\texttt{CoreSet}(\Phi) := (\mathcal{K}, \rho) \quad \text{where} \quad \mathcal{K} \subset \mathcal{A} \times \mathcal{B},\ \rho \in \Delta_{\mathcal{K}}, \quad \text{such that}$$
$$\max_{\phi \in \Phi} \phi^\top \Big(\sum_{(a,b) \in \mathcal{K}} \rho(a, b) \phi(a, b) \phi(a, b)^\top \Big)^{-1} \phi \le 2d \quad \text{and} \quad K = |\mathcal{K}| \le 4d \log\log d + 16. \tag{10}$$

Such a core set is guaranteed to exist for any finite Φ [23, Theorem 4.4], and can be found efficiently in $O(ABd^2)$ steps of computation [42, Lemma 3.9]. (2) Rather than estimating the reward at every (a, b) in a tabular fashion, we use a weighted least-squares (9) to obtain estimates $(\widehat{\theta}_1, \widehat{\theta}_2)$ which are then used to approximate the true reward for all (a, b).

We now state our main guarantee for Algorithm 3. The proof can be found in Appendix E.1.

Theorem 5 (Learning Stackelberg in linear bandit games)**.** *For any linear bandit game, Algorithm 3 outputs a* $(\mathsf{gap}_\varepsilon + \varepsilon)$*-approximate Stackelberg equilibrium* $(\widehat{a}, \widehat{b})$ *with probability at least* $1 - \delta$*:*

$$\phi_0(\widehat{a}) \ge \phi_{\varepsilon/2}(\widehat{a}) \ge \max_{a \in \mathcal{A}} \phi_0(a) - \mathsf{gap}_\varepsilon - \varepsilon,$$

in at most $n = \widetilde{O}(d^2/\varepsilon^2)$ *queries.*

Sample complexity; computation Theorem 5 shows that Algorithm 3 achieves $\widetilde{O}(d^2/\varepsilon^2)$ sample complexity for learning Stackelberg equilibria in linear bandit games. This only depends polynomially on the feature dimension d instead of the size of the action spaces A, B, which improves over Algorithm 1 when A, B are large and is desired given the linear structure (8). This sample complexity has at most a $\widetilde{O}(d)$ gap from the lower bound: An $\Omega(d/\varepsilon^2)$ lower bound for linear bandit games can be obtained by directly adapting $\Omega(AB/\varepsilon^2)$ lower bound for bandit games in Theorem 3 (see Appendix E.2 for a formal statement and proof). We also note that, while the focus of Theorem 5 is on the sample complexity rather than the computation, Algorithm 3 is guaranteed to run in $\mathrm{poly}(A, B, d, 1/\varepsilon^2)$ time, since the `CoreSet` subroutine, the weighted least squares step (9), and the final optimization step in approximate best-response sets can all be solved in polynomial time.

6 Conclusion

This paper provides the first line of sample complexity results for learning Stackelberg equilibria in general-sum games. We identify a fundamental gap between the exact and estimated versions of the Stackelberg value, and design sample-efficient algorithms for learning Stackelberg with value optimal up to this gap, in several representative two-player general-sum games. Our results open up a number of interesting future directions, such as optimal sample complexity in bandit-RL games and linear-bandit games, as well as learning Stackelberg in more general Markov games.

References

[1] A. Ahmadinejad, S. Dehghani, M. Hajiaghayi, B. Lucier, H. Mahini, and S. Seddighin. From duels to battlefields: Computing equilibria of blotto and other games. *Mathematics of Operations Research*, 44(4):1304–1325, 2019.

[2] E. Altman. *Constrained Markov decision processes*, volume 7. CRC Press, 1999.

[3] M. G. Azar, I. Osband, and R. Munos. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*, pages 263–272. PMLR, 2017.

[4] Y. Bai and C. Jin. Provable self-play algorithms for competitive reinforcement learning. In *International Conference on Machine Learning*, pages 551–560. PMLR, 2020.

[5] Y. Bai, C. Jin, and T. Yu. Near-optimal reinforcement learning with self-play. *arXiv preprint arXiv:2006.12007*, 2020.

[6] A. Blum, N. Haghtalab, and A. D. Procaccia. Learning optimal commitment to overcome insecurity. 2014.

[7] A. Blum, N. Haghtalab, M. Hajiaghayi, and S. Seddighin. Computing stackelberg equilibria of large general-sum games. In *International Symposium on Algorithmic Game Theory*, pages 168–182. Springer, 2019.

[8] S. P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[9] N. Cesa-Bianchi and G. Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[10] R. Cole and T. Roughgarden. The sample complexity of revenue maximization. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 243–252, 2014.

[11] V. Conitzer and T. Sandholm. Complexity of mechanism design. *arXiv preprint cs/0205075*, 2002.

[12] V. Conitzer and T. Sandholm. Self-interested automated mechanism design and implications for optimal combinatorial auctions. In *Proceedings of the 5th ACM Conference on Electronic Commerce*, pages 132–141, 2004.

[13] V. Conitzer and T. Sandholm. Computing the optimal strategy to commit to. In *Proceedings of the 7th ACM conference on Electronic commerce*, pages 82–90, 2006.

[14] P. Dütting, Z. Feng, H. Narasimhan, D. Parkes, and S. S. Ravindranath. Optimal auctions through deep learning. In *International Conference on Machine Learning*, pages 1706–1715. PMLR, 2019.

[15] T. Fiez, B. Chasnov, and L. J. Ratliff. Convergence of learning dynamics in stackelberg games. *arXiv preprint arXiv:1906.01217*, 2019.

[16] D. J. Foster, Z. Li, T. Lykouris, K. Sridharan, and E. Tardos. Learning in games: Robustness of fast convergence. *arXiv preprint arXiv:1606.06244*, 2016.

[17] C. Jin, Z. Allen-Zhu, S. Bubeck, and M. I. Jordan. Is q-learning provably efficient? *arXiv preprint arXiv:1807.03765*, 2018.

[18] C. Jin, A. Krishnamurthy, M. Simchowitz, and T. Yu. Reward-free exploration for reinforcement learning. *arXiv preprint arXiv:2002.02794*, 2020.

[19] C. Jin, P. Netrapalli, and M. Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? In *International Conference on Machine Learning*, pages 4880–4889. PMLR, 2020.

[20] G. Korpelevich. The extragradient method for finding saddle points and other problems. *Matecon*, 12:747–756, 1976.

[21] D. Korzhyk, Z. Yin, C. Kiekintveld, V. Conitzer, and M. Tambe. Stackelberg vs. nash in security games: An extended investigation of interchangeability, equivalence, and uniqueness. *Journal of Artificial Intelligence Research*, 41:297–327, 2011.

[22] T. Lattimore and C. Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[23] T. Lattimore, C. Szepesvari, and G. Weisz. Learning with good feature representations in bandits and in rl with a generative model. In *International Conference on Machine Learning*, pages 5662–5670. PMLR, 2020.

[24] J. Z. Leibo, V. Zambaldi, M. Lanctot, J. Marecki, and T. Graepel. Multi-agent reinforcement learning in sequential social dilemmas. *arXiv preprint arXiv:1702.03037*, 2017.

[25] J. Letchford and V. Conitzer. Computing optimal strategies to commit to in extensive-form games. In *Proceedings of the 11th ACM conference on Electronic commerce*, pages 83–92, 2010.

[26] J. Letchford, V. Conitzer, and K. Munagala. Learning and approximating the optimal strategy to commit to. In *International Symposium on Algorithmic Game Theory*, pages 250–262. Springer, 2009.

[27] Q. Liu, T. Yu, Y. Bai, and C. Jin. A sharp analysis of model-based reinforcement learning with self-play. *arXiv preprint arXiv:2010.01604*, 2020.

[28] A. Marchesi, F. Trovò, and N. Gatti. Learning probably approximately correct maximin strategies in simulation-based games with infinite strategy spaces. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, pages 834–842, 2020.

[29] J. Nash. Non-cooperative games. *Annals of mathematics*, pages 286–295, 1951.

[30] A. Nemirovski. Prox-method with rate of convergence o (1/t) for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004.

[31] A. S. Nemirovski and D. B. Yudin. Cesari convergence of the gradient method of approximating saddle points of convex-concave functions. In *Doklady Akademii Nauk*, volume 239, pages 1056–1059. Russian Academy of Sciences, 1978.

[32] M. Nouiehed, M. Sanjabi, J. D. Lee, and M. Razaviyayn. Solving a class of non-convex min-max games using iterative first order methods. *arXiv preprint arXiv:1902.08297*, 2019.

[33] B. Peng, W. Shen, P. Tang, and S. Zuo. Learning optimal strategies to commit to. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2149–2156, 2019.

[34] J. Pérolat, F. Strub, B. Piot, and O. Pietquin. Learning nash equilibrium for general-sum markov games from batch data. In *Artificial Intelligence and Statistics*, pages 232–241. PMLR, 2017.

[35] H. Rafique, M. Liu, Q. Lin, and T. Yang. Non-convex min-max optimization: Provable algorithms and applications in machine learning. *arXiv preprint arXiv:1810.02060*, 2018.

[36] A. Rakhlin and K. Sridharan. Optimization, learning, and games with predictable sequences. *arXiv preprint arXiv:1311.1869*, 2013.

[37] T. Roughgarden. Algorithmic game theory. *Communications of the ACM*, 53(7):78–86, 2010.

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

[39] Y. Shoham and K. Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.

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

[41] M. Tambe. *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge university press, 2011.

[42] M. J. Todd. *Minimum-volume ellipsoids: Theory and algorithms*. SIAM, 2016.

[43] J. von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320, 1928.

[44] B. Von Stengel and S. Zamir. Leadership games with convex strategy sets. *Games and Economic Behavior*, 69(2):446–457, 2010.

[45] M. J. Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

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

[47] K. Zhang, S. M. Kakade, T. Başar, and L. F. Yang. Model-based multi-agent rl in zero-sum markov games with near-optimal sample complexity. *arXiv preprint arXiv:2007.07461*, 2020.

[48] S. Zheng, A. Trott, S. Srinivasa, N. Naik, M. Gruesbeck, D. C. Parkes, and R. Socher. The ai economist: Improving equality and productivity with ai-driven tax policies. *arXiv preprint arXiv:2004.13332*, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] As this is a theoretical paper, we do not foresee any immediate negative societal impact of this work.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]