Learning in Multi-Player Stochastic Games

William Brown[1]

[1]Computer Science Dept., Columbia University, New York, New York, USA

Preliminary version

Abstract

We consider the problem of simultaneous learning in stochastic games with many players in the finite-horizon setting. While the typical target solution for a stochastic game is a Nash equilibrium, this is intractable with many players. We instead focus on variants of *correlated equilibria*, such as those studied for extensive-form games. We begin with a hardness result for the adversarial MDP problem: even for a horizon of 3, obtaining sublinear regret against the best non-stationary policy is NP-hard when both rewards and transitions are adversarial. This implies that convergence to even the weakest natural solution concept—normal-form coarse correlated equilbrium—is not possible via black-box reduction to a no-regret algorithm even in stochastic games with constant horizon (unless $\mathsf{NP} \subseteq \mathsf{BPP}$). Instead, we turn to a different target: algorithms which *generate* an equilibrium when they are used by all players. Our main result is algorithm which generates an *extensive-form* correlated equilibrium, whose runtime is exponential in the horizon but polynomial in all other parameters. We give a similar algorithm which is polynomial in all parameters for "fast-mixing" stochastic games. We also show a method for efficiently reaching normal-form coarse correlated equilibria in "single-controller" stochastic games which follows the traditional no-regret approach. When shared randomness is available, the two generative algorithms can be extended to give simultaneous regret bounds and converge in the traditional sense.

1 INTRODUCTION

Many multi-agent systems, such as financial markets, transportation networks, and video games, involve agents competing in environments where their actions affect their immediate rewards as well as transitions between states in the environment. When opponent strategies are fixed, this resembles a reinforcement learning problem for a single agent. Stochastic games, also known as Markov games are a popular model for multi-agent reinforcement learning problems (Littman [1994]), and have also been studied extensively throughout economics and computer science (Solan and Vieille [2015], Shoham and Leyton-Brown [2008]). They generalize Markov decision processes (MDPs) to many players, where each state is now a game where both the instantaneous rewards and transitions depend on the actions of all players. As is the case throughout game theory, a fundamental question from the perspective of algorithm design is whether some kind of equilibrium can be found efficiently.

The traditional solution concept for a game, often interpreted as a model of rational behavior, is the Nash equilibrium (Nash [1950]). In two-player zero-sum and other restricted classes of normal-form games, Nash equibria can be found efficiently; however, finding one is PPAD-complete for arbitrary games even with only two players (Daskalakis et al. [2006], Chen et al. [2007]), and thus likely computationally intractable. A more appropriate target in this case is a *correlated* equilibrium, introduced by Aumann [1974], which is a generalization of a Nash equilibrium where strategies can be correlated across players, and can be efficiently computed in general games (e.g. Nisan et al. [2007]). Correlated equilibria can also be reached through repeated play by agents who use appropriate learning algorithms. The existence of no-swap-regret dynamics which efficiently converge to correlated equilibria is a celebrated result in the theory of learning in games (Foster and Vohra [1997], Hart and Mas-Colell [2000], Blum and Mansour [2004]). A notable benefit of this approach is that it does not depend on the description length of the game, so long as rewards are computable from an action profile, and thus can be used in many-player games where writing an explicit game description is prohibitive.

The normal-form game model is often insufficient to capture problems of practical interest. The aforementioned results

Accepted for the 37th Conference on Uncertainty in Artificial Intelligence (UAI 2021).

cannot be applied directly to stochastic games, as the strategy space is exponential in the relevant parameters. Yet, as real-world problems often have many players and possibly arbitrary reward structures, it is natural to target correlated equilibria as a solution concept for stochastic games as well. The starting point for our work is asking whether an efficient convergence result of the same form as Hart and Mas-Colell [2000] can be obtained for repeated play of a stochastic game in the finite-horizon setting.

A related setting where similar questions have been studied is that of extensive-form games (EFGs). Several refinements of correlated equilibria have been proposed for EFGs, which differ in when action recommendations are revealed to each agent (von Stengel and Forges [2008], Huang and von Stengel [2008], Farina et al. [2019]). Two variants which we will consider are normal-form coarse correlated equilibria (NFCCE) and extensive-form correlated equilibria (EFCE), with the latter contained in the former, which we adapt to finite-horizon stochastic games. Recent work has led to the development of an algorithm which converges to an EFCE by minimizing an appropriate notion of regret for each agent (Celli et al. [2020]). We show that such a black-box reduction cannot work for stochastic games of even constant horizon, as the corresponding online learning problem is hard, and instead design algorithms which converge to correlated equilibria (in a somewhat delicate sense) by directly leveraging information about opponents' strategies.

1.1 RESULTS AND TECHNIQUES

We assume that players in a finite-horizon stochastic game play for many repeated horizons, or *trajectories*, and that rewards and transition dynamics are computed by an oracle when players submit actions simultaneously at a given state. Players receive only *bandit* feedback, i.e. they do not know what see what rewards or transitions would have occurred if they had selected a different action. Longer horizons allow for greater consideration of "deferred rewards" for actions, such as in a board game where an early move can become consequential in the endgame; a horizon of one corresponds to a repeated one-shot game. Each form of correlated equilibrium we consider is a joint distribution over recommended *policies*, which tell each player an action to play at each state. We consider policies which are non-stationary, i.e. they can depend on the time-step. In a NFCCE, no player can improve their reward by committing to a fixed policy before the trajectory begins or recommendations are revealed. In an EFCE, players receive individual action recommendations only upon reaching a state, and they cannot improve rewards by "swapping" their actions based on their recommendations.

Our first result is negative: we show that obtaining sublinear regret against the best non-stationary policy for adversarial MDPs with a horizon of 3 is NP-hard, strengthening previous hardness results (Even-Dar et al. [2004], Abbasi-Yadkori et al. [2013]) which require the stronger "LPN hardness" assumption and hold only when the horizon is approximately the size of the MDP. The adversarial MDP problem is the natural online learning variant of our setting, as each set of opponent policies defines an MDP for a given player, albeit with different rewards and transitions. Assuming $\mathsf{NP} \not\subseteq \mathsf{BPP}$, this implies that any algorithm which quickly converges to even a NFCCE in a stochastic game with constant horizon cannot be no-regret against arbitrary opponents, ruling out a black-box reduction to reaching a correlated equilibrium as in Hart and Mas-Colell [2000] or Celli et al. [2020].

We then turn our attention to designing algorithms which make use of information about the behavior of opponents, namely that they are using the same algorithm. While regret minimization and learning equilibria are often viewed as intimately connected, lower bounds for regret do not necessarily imply barriers for equilibria when opponents are not behaving arbitrarily; in particular, knowledge of "self-play" has been used to obtain rates of convergence to correlated equilibria in normal-form games which overcome lower bounds for regret minimization against an arbitrary adversary (Syrgkanis et al. [2015], Chen and Peng [2020]).

Our main result is a decentralized learning algorithm which reaches an EFCE when used by all players, and in particular one where the distribution of recommended action profiles at each state is a product distribution across states. We observe that *computing* an EFCE of this form is straightforward in a centralized model, as it reduces to the problem of finding a correlated equilibrium for a set of normal-form games, each of which can be computed with linear programming or no-swap-regret learning. States at the final time-step are essentially equivalent to normal-form games, and each player will have a *value* associated with a given correlated equilibrium representing their average reward at that state-time pair. These values can be folded back into rewards at previous time-steps, enabling an inductive computation. Our main algorithm, PLL, aims to simulate this approach by conducting repeated *parallel local learning* at each state. After a number of trajectories which is exponential in the horizon length but polynomial in all other parameters, the set of *subgame value estimates* stabilizes for each player, at which point the product distribution across state-time pairs over the action profiles generated by continued local learning constitutes an EFCE for the stochastic game, thus circumventing the previous hardness result. In addition, we give a variant of PLL which removes the exponential dependence on horizon provided that a "mixing" assumption is satisfied.

We also give an alternative approach which reaches an NFCCE in "single-controller" stochastic games, where only one player affects transitions (as studied in e.g. Filar and Raghavan [1984]). Here, the controller uses a no-regret algorithm for adversarial MDPs with fixed transitions (Rosen-

Preliminary version

berg and Mansour [2019]) while the followers use another variant of PLL. This approach converges in the black-box sense, where each agent has sublinear regret for the the uniform distribution over the entire history of strategies. We further show that the algorithms for general and fast-mixing stochastic games can be extended to satisfy sublinear regret bounds simultaneously for all agents if shared randomness is available by allowing agents to play according to the generated equilibrium after the initial algorithms terminate. As building blocks for the analysis of our algorithms, we establish generalizations of known results for convergence of learning algorithms to correlated equilibria in normal-form games (e.g. Blum and Mansour [2004], to the case where reward feedback is noisy, which we call "games with stochastic rewards") and Bayesian games (removing the "independent private value" assumption in Hartline et al. [2015]). Most proofs and some algorithmic details (such as exact constants) are deferred to Appendix A.

1.2 COMPARISON WITH RELATED WORK

Most provably efficient algorithms for learning in stochastic games target Nash equilibria in tractable special cases like zero-sum games, and often in infinite-horizon settings with discount factors or mixing guarantees (Brafman and Tennenholtz [2001], Chang et al. [2010], Zhang et al. [2018], Zhang et al. [2018]). When there are many players, we cannot afford to "learn the game" and use a model-based approach (e.g. Brafman and Tennenholtz [2001]), as explicitly representing even a single state will be intractable. Closest to our setting is Kearns et al. [2000], who give a centralized recursive algorithm that *computes* an EFCE in finite-horizon stochastic games for the case when the algorithm can sample many transitions and rewards at each state (which we cannot do in our "repeated trajectories" model); the runtime is exponential in both the horizon and the number of players, but does not depend on the number of states. Correlated equilibria are also studied empirically by Greenwald and Hall [2003], and there is a large body of literature on general-sum multi-agent learning under other objectives or without convergence guarantees; for a recent overview of multi-agent reinforcement learning, see Zhang et al. [2019].

Finite-horizon stochastic games are somewhat related to extensive-form games, but are distinct in several important ways and in general are not directly comparable. In EFGs, only one player acts at each state, but partial information is allowed via "infosets", which can be used to simulate simultaneous actions (Shoham and Leyton-Brown [2008], von Stengel and Forges [2008], Celli et al. [2020]). Stochastic games with partial information have been considered in the literature (Hansen et al. [2004]), but are considerably more difficult to solve (POMDPs, the single-player analog, are PSPACE-complete, see Papadimitriou and Tsitsiklis [1987]), and we will not consider them here. EFGs typically enforce a tree structure on the infosets by the "perfect recall" assumption, whereas finite-horizon stochastic games allow for a DAG structure. As a result, our setting can allow for games with both a large depth and branching factor as long as the number of total states is not too large; EFGs are not as appropriate of a model when there are many paths to a given game state. Encoding a finite-horizon stochastic game as an EFG requires considering each path to a state independently, introducing a space blowup which is exponential in the horizon length, which renders existing methods for learning in EFGs impractical for our setting.

2 CORRELATED EQUILIBRIA IN STOCHASTIC GAMES

We begin with some background regarding no-(swap)-regret learning in the bandit feedback setting and connections to correlated equilibria in Section 2.1. In Section 2.2, we introduce a preliminary model of a "game with stochastic rewards" and its corresponding definition of a correlated equilibrium. This serves as a building block for our formulation of stochastic games in Section 2.3. These game models may have unbounded description length; throughout, we treat them as oracles to which players submit actions simultaneously, then receive reward and state feedback. We assume instantaneous rewards are normalized to lie in $[0, 1]$.

2.1 PRELIMINARIES

Adversarial Bandits. In the *adversarial multi-armed bandit* problem, the objective is to sequentially choose actions $a \in \mathcal{A}$ which minimize some notion of *regret*, where rewards at each step are chosen by an (adaptive) adversary. Let N denote the cardinality of $\mathcal{A}$. At each round, an algorithm commits to a distribution of actions $q^t \in \Delta(\mathcal{A})$, which is observed by an adversary, who then chooses a reward vector $b^t \in [0, 1]^N$. The algorithm then draws an action a^t from q^t and then observes the associated reward $b^t_{a^t}$. Let $\mathcal{F}$ denote the set of *swap functions* $\mathcal{F} : \mathcal{A} \to \mathcal{A}$. After T rounds, the *swap-regret* of such an algorithm is given by

$$\text{Reg}^{\mathcal{F}}(T) = \max_{f \in \mathcal{F}} \sum_{t=1}^{T} b^t_{f(a^t)} - \sum_{t=1}^{T} b^t_{a^t}.$$

Dividing by T gives us the *average* swap regret; there are efficient algorithms for achieving sublinear swap-regret in this setting, which we refer to as *no-swap-regret* as average swap regret vanishes as T grows.

Proposition 1 (Blum and Mansour [2004])**.** *There is an algorithm (*SR-MAB*) that, when used for T rounds in the multi-armed bandit setting with adaptively chosen losses, has expected swap regret bounded by $O(N\sqrt{NT \log N})$.*

This implies that after using SR-MAB for $O(\frac{1}{\epsilon^2} N^3 \log N)$ rounds, the expected average swap regret is bounded by ϵ.

Preliminary version

We will use $\mathcal{B}$ to denote the SR-MAB algorithm and $B(\epsilon)$ to denote the number of rounds after which it has expected average swap regret at most ϵ. We use it as a subroutine in our algorithms, but our results are not specific to its details, and can be adapted to use any no-swap-regret algorithm.

Correlated Equilibria in Normal-Form Games. In a normal-form game with M players and action space $\mathcal{A} = \times_{i\in[M]} \mathcal{A}_i$, each player i selects an action $a_i \in \mathcal{A}_i$ and receives a reward given by a utility function $u : \mathcal{A} \to [0,1]^M$ mapping action profiles to a vector of rewards. An ϵ-*correlated equilibrium* for such a game is a distribution $D \in \Delta(\mathcal{A})$ such that for all players i and deterministic functions $f : \mathcal{A}_i \to \mathcal{A}_i$,

$$\mathop{\mathbb{E}}_{a\sim D}[u_i(a_i; a_{-i})] \geq \mathop{\mathbb{E}}_{a\sim D}[u_i(f(a_i); a_{-i})] - \epsilon,$$

i.e. no player can benefit by more than ϵ in expectation by deviating from the distribution of "recommended actions" with any swap function. Repeated play in a normal form game converges to a correlated equilibrium if players use no-swap-regret algorithms.

Proposition 2 (Blum and Mansour [2004])**.** *If all players in a game select actions using $\mathcal{B}$ for $B(\epsilon)$ rounds, the uniform distribution over the sequence of action profiles played thus far is an ϵ-correlated equilibrium for the game, where the expectation is taken both with respect to the distribution of action profiles as well as the randomness of $\mathcal{B}$.*

2.2 GAMES WITH STOCHASTIC REWARDS

We define a *game with stochastic rewards* as a distribution over normal-form games, which is equivalent to a normal-form game where reward feedback can be noisy and arbitrarily correlated across players.

Definition 1 (Games with Stochastic Rewards)**.** *A game with stochastic rewards $x = (\mathcal{A}, M, r, u)$ with M players is given by a set of action profiles $\mathcal{A} = \times_{i\in[M]} \mathcal{A}_i$, a distribution over reward tensors $r \in \Delta(\Theta)$, and a utility function u, where the utilities $u : \mathcal{A} \times \Theta \to [0,1]^M$ depend on the realization of $\theta \sim r$. In a round of the game, players submit actions to x simultaneously, θ is drawn independently from r, and each player observes only their utility $u_i(a, \theta)$.*

We assume that $\mathcal{A}_i = N$ for all agents. A correlated equilibrium for such a game is an action profile distribution where the regret bound holds with respect to the distribution over reward tensors.

Definition 2 (Correlated Equilibria in Games with Stochastic Rewards)**.** *An ϵ-correlated equilibrium for a game with stochastic rewards is a distribution $D \in \Delta(\mathcal{A})$ such that for all players and all swap functions $f \in \mathcal{F}$,*

$$\mathop{\mathbb{E}}_{a\sim D,\theta\sim r}[u_i(a_i; a_{-i}, \theta)] \geq \mathop{\mathbb{E}}_{a\sim D,\theta\sim r}[u_i(f(a_i); a_{-i}, \theta)] - \epsilon.$$

Here, there is some expected reward tensor $\bar{\theta}$ where for every action profile a and player i, $\bar{\theta}_{a,i} = \mathbb{E}_{\theta\sim r}[u_i(a_i; a_{-i}, \theta)]$, and a correlated equilibrium for such a game is simply a correlated equilibrium for the game specified by $\bar{\theta}$.

2.3 FINITE-HORIZON STOCHASTIC GAMES

Stochastic games resemble Markov decision processes, yet there are many players who act simultaneously at each state, and transition and reward dynamics depend on all players' actions. In finite-horizon stochastic games, players begin at a state drawn from some initial distribution and play for a fixed period of steps, where a step consists of one set of simultaneous actions, followed by a transition to a new state and a reward for each agent. We allow both rewards and transitions to be probabilistic. A *trajectory* is the sequence of steps over the entire horizon length.

Definition 3 (Finite-Horizon Stochastic Games)**.** *A finite-horizon stochastic game is given by a tuple $\mathcal{M} = (\mathcal{X}, \mathcal{A}, M, H, p_0, p, r, u)$, where:*

- *M is the number of players,*
- *$\mathcal{A} = \times_{i\in[M]} \mathcal{A}_i$ is the action space ($|\mathcal{A}_i| = N$ for each player),*
- *H is the horizon length,*
- *$\mathcal{X}$ is the state space ($|\mathcal{X}| = S$),*
- *$p_0 \in \Delta(\mathcal{X})$ is an initial distribution over states,*
- *$p : [H] \to \Delta(\mathcal{T})$ is a function which defines distributions over transition functions $\tau \in \mathcal{T} : \mathcal{A} \times \mathcal{X} \to \mathcal{X} \cup \varnothing$,*
- *$r : \mathcal{X} \times H \to \Delta(\Theta)$ is a function which defines distributions of reward tensors, and*
- *$u : \mathcal{A} \times \Theta \to [0,1]^M$ is a function which defines utilities for each player given an action profile and a reward tensor.*

For all a, x, τ, we assume $\tau(a, x) = \varnothing$ if and only if $h = H$, where $\varnothing$ denotes termination of the episode.

State Values for a Policy Profile. We consider non-stationary policies of the form $\pi_i : \mathcal{X} \times [H] \to \mathcal{A}_i$ for each agent i, with $\pi_i \in \Pi_i$ and $\Pi = \times_{i\in[M]} \Pi_i$. For a policy profile π we can recursively define a state value function $V_i^\pi : \mathcal{X} \times [H] \to [0, H]$ where:

$$V_i^\pi(x, H) = \mathop{\mathbb{E}}_{\theta\sim r(x,H)}[u_i(a_i; a_{-i}, \theta)]$$

where $a_i = \pi_i(x, h)$ for each agent i and

$$V_i^\pi(x, h) = \mathop{\mathbb{E}}_{\theta,\tau}[u_i(a_i; a_{-i}, \theta) + V_i^\pi(\tau(a, x), h + 1)]$$

for $h \in \{1, \ldots, H-1\}$, where θ and τ are drawn from the appropriate distributions.

Preliminary version

Counterfactual State Values. We also define the *counterfactual* state value function for a player who deviates from a distribution over policy profiles. We consider two kinds of deviations: always playing a fixed policy $\psi_i \in \Pi_i$, or deviating from local recommendations using a "swap function". Let $\mathcal{F}_i : \mathcal{A}_i \times \mathcal{X} \times [H] \to \mathcal{A}_i$ be the set of swap functions for player i that can depend on action, state, and episode step. We can recursively define a value function for $f \in \mathcal{F}_i$ given a policy profile π:

$$V_i^{\pi,f}(x, H) = \mathop{\mathbb{E}}_{\theta \sim r(x,H)}[u_i(f(a_i, x, h); a_{-i}, \theta)]$$

and

$$V_i^{\pi,f}(x, h) = \mathop{\mathbb{E}}_{\theta,\tau}[u_i(f(a_i, x, h); a_{-i}, \theta) + V_i^{\pi,f}(\tau^*, h + 1)],$$

where $\tau^* = \tau(f(a_i, x, h), a_{-i}, x)$ and again $a_i = \pi_i(x, h)$ for each agent i. Our notion of swap regret will be defined with respect to $\mathcal{F}_i$, and our notion of a correlated equilibrium is a distribution over policy profiles $\pi = [\pi_i]_{i \in [M]}$. We can equivalently define $V_i^{\pi,\psi_i}(x, h)$ for $\psi_i \in \Pi_i$, omitting dependence on the actions recommended at each step.

We can adapt variants of correlated equilibria as considered for extensive-form games in e.g. von Stengel and Forges [2008] or Farina et al. [2019] to stochastic games. Our definition of a normal-form correlated equilibrium says that no player can benefit substantially by committing to a fixed policy before seeing any recommendations, given knowledge of a policy profile distribution. For an extensive-form correlated equilibrium, recommendations are revealed to agents one step at a time, and they cannot benefit by deviating from these recommendations using a swap function. The EFCEs we consider will be a product distribution across state-time pairs, and so we restrict to considering deviations based only on the current recommendation—recommendations at previous steps provide no additional information about opponent recommendations at any other step.

Definition 4 (Normal-Form Coarse Correlated Equilibria for Stochastic Games)**.** *We say that a policy profile distribution $D \in \Delta(\Pi)$ is an ϵ-approximate normal-form coarse correlated equilibrium (or ϵ-NFCCE) for a finite-horizon stochastic game if for all agents i and all $\psi_i \in \Pi_i$:*

$$\mathop{\mathbb{E}}_{x \sim p_0, \pi \sim D}[V_i^{\pi}(x, 1)] \geq \mathop{\mathbb{E}}_{x \sim p_0, \pi \sim D}[V_i^{\pi,\psi}(x, 1)] - \epsilon H.$$

Definition 5 (Extensive-Form Correlated Equilibria for Stochastic Games)**.** *We say that a policy profile distribution $D \in \Delta(\Pi)$ is an ϵ-approximate extensive-form correlated equilibrium (or ϵ-EFCE) for a finite-horizon stochastic game if for all agents i and all $f \in \mathcal{F}_i$:*

$$\mathop{\mathbb{E}}_{x \sim p_0, \pi \sim D}[V_i^{\pi}(x, 1)] \geq \mathop{\mathbb{E}}_{x \sim p_0, \pi \sim D}[V_i^{\pi,f}(x, 1)] - \epsilon H.$$

Just as in the case for extensive-form games, EFCEs provide stronger guarantees than NFCCEs.

Theorem 1. *For a finite-horizon stochastic game, the set of ϵ-EFCEs is contained in the set of ϵ-NFCCEs for all $\epsilon \geq 0$.*

Proof. Any fixed policy ψ can be encoded with the swap function $f(a_i, x, h) = \psi(x, h)$ for all a_i, x, and h, and so any ϵ-EFCE is also an ϵ-NFCCE. □

These definitions bound the average per-step regret by ϵ for each agent under the appropriate class of deviations. We are interested in when, and how quickly, players can converge to such an equilibrium by repeatedly playing the game.

3 HARDNESS OF LEARNING IN ADVERSARIAL MDPS

The first thing that one might hope for in the setting of multi-player finite-horizon stochastic games is the existence of an algorithm which minimizes the appropriate notion of regret for each agent, and can be used as a *black box* to reach a correlated equilibrium. This is the form of Celli et al. [2020], who give an algorithm with sublinear *trigger regret*, which corresponds to the definition of an EFCE and thus results in efficient convergence of the sequence of policies played to an EFCE when all agents use the algorithm.

The appropriate problem for modeling repeated play in finite-horizon stochastic games against arbitrary opponents is the "adversarial MDP problem", where an agent is faced with a set of finite-horizon MDPs, each with a different reward and transition function. A commonly studied objective is to minimize regret against the best fixed policy, and such an algorithm with sublinear regret would converge to an NFCCE for a stochastic game. This was shown to be as hard as the "learning parities with noise" problem (which is not known to be NP-hard) by Abbasi-Yadkori et al. [2013] when $H = \Theta(S)$. We show that this is indeed NP-hard even when the horizon is only 3.

Theorem 2. *Assuming* NP $\not\subseteq$ BPP*, there is no algorithm with polynomial time per-round computation which has $O(T^{1-\delta} \cdot \text{poly}(S))$ regret algorithm for the adversarial MDP problem with $H \geq 3$, for any $\delta > 0$.*

We prove this by considering an offline version of the adversarial MDP problem, where the goal is to find a single non-stationary policy which does well across a set of MDPs with differing reward and transition functions. We show that this is as hard as MAX-3-SAT, and use the online-to-batch reduction from Cesa-Bianchi et al. [2004] to show hardness of the online problem. This suggests we should not expect a black-box reduction to finding even an NFCCE, even when the horizon is quite short.

Corollary 2.1. *Assuming* NP $\not\subseteq$ BPP*, any decentralized learning algorithm which converges in polynomial time to an approximate (coarse) correlated equilibrium for a*

Preliminary Version

stochastic game with horizon $H \geq 4$ when used by all players must have regret $\omega(T^{1-\delta} \cdot \text{poly}(S))$, for all $\delta > 0$, against arbitrary opponents.

Despite this, we give algorithms which converge to correlated equilibria which are not black-box, i.e. they explicitly make use of the fact that all players are using the same algorithm. Without any assumptions on transitions, the runtime of our primary algorithm, PLL, is polynomial in all parameters except the horizon, where dependence is exponential in the worst case, allowing us to overcome the barrier we show for black-box reductions.

4 LEARNING IN STOCHASTIC GAMES VIA REPEATED TRAJECTORIES

The main idea behind our algorithm is for each agent to locally perform no-swap-regret learning at each state-time pair, augmenting their observed rewards with estimates of the "values" for states they transition to. We first give an extension of the convergence theorem for bandit learning in normal-form games from Blum and Mansour [2004] to "games with stochastic rewards", which makes use of $\mathcal{B}$ with additional modifications in order to handle stochasticity and obtain high-probability bounds for both regret and value estimates. We then give an "offline" centralized algorithm, BILL, which uses this subroutine to *compute* an EFCE for a stochastic game, given the ability to sample rewards and transitions for each state. Our algorithm PLL can be viewed as simulating BILL in a decentralized manner when agents play repeated trajectories of the game. The sense in which PLL converges is different from e.g. Blum and Mansour [2004]; rather than taking the uniform distribution over the history of policies, we consider the product distribution of a truncated history of action profiles at each state-time pair. We can improve the speed of convergence for PLL when a "fast-mixing" assumption is satisfied, a common tool in the analysis of reinforcement learning algorithms.

4.1 LEARNING IN GAMES WITH STOCHASTIC REWARDS

Recall that we define correlated equilibria for stochastic games with respect to the average reward tensor $\bar{\theta}$. When agents all use a no-swap regret algorithm (such as $\mathcal{B}$) to play such a game repeatedly, the immediate regret bound holds with respect to the realized sequence of reward tensors. We can extend this bound to hold with respect to $\bar{\theta}$ by viewing the "error" of each swap function for a player (their reward from sampled sequence of reward tensors θ versus the average tensor $\bar{\theta}$) as a martingale which does not deviate too far from its expectation. Depending on the relationship between ϵ and N, we may need to run $\mathcal{B}$ for slightly longer than $B(\epsilon)$ in order to apply our martingale analysis, but only by at most a factor of $O(\log(1/\epsilon))$. We let $B(\epsilon, N)$ denote this extended runtime as a function of ϵ and N.

Theorem 3. *When players in a game with stochastic rewards x select actions using $\mathcal{B}$ for $T \geq B(\epsilon/4, N)$ rounds, the sequence of action profiles is an ϵ-correlated equilibrium for the game, where the expectation is taken with respect to the tensor distribution as well as $\mathcal{B}$.*

The proof is given in Appendix A.2. By running $\mathcal{B}$ several times, we can boost the expected regret bound for each player to hold with high probability over the randomness of while simultaneously obtaining accurate estimates of the *value* of this process for each player; we use this form of the result in the analysis for later algorithms.

Corollary 3.1. *When all agents in a game with stochastic rewards x play according to $\mathcal{B}$ for at least $\frac{2\log(5M/\delta)}{\eta^2} \cdot B(\epsilon/8, N)$ rounds, simultaneously restarting $\mathcal{B}$ every $B(\epsilon/8, N)$ rounds, the resulting sequence of actions is an $(\epsilon/2 + \eta/2)$-correlated equilibrium for x with probability at least $1 - \delta/5$.*

Further, let $V_i^{\mathcal{B}}(x) = \mathbb{E}_{\mathcal{B},r}\left[\frac{1}{T}\sum_{t=1}^{T} u_i(a_i^t; a_{-i}^t, \theta^t)\right]$ and let $\hat{V}_i^{\mathcal{B}}(x)$ be the average utility received by player i over all rounds. With probability at least $1 - 2\delta/5$, $\left|V_i^{\mathcal{B}}(x) - \hat{V}_i^{\mathcal{B}}(x)\right| \leq \eta/2$ simultaneously for all players.

Additionally, the computed estimate is within η of player i's expected average reward for playing the game according to the resulting policy distribution with probability at least $1 - 2\delta/5$.

An extension of this method to Bayesian games is presented in Appendix A.3, which we make use of in analyzing Algorithm 4 (Theorem 7).

4.2 SUBGAME VALUE ESTIMATES

We define a notion of the *subgame value* for an agent at a state-step pair (x, h) in a stochastic game, similiar to that in Definition 5, which is specified with respect to a learning algorithm $\mathcal{B}$. Henceforth we will refer to (x, h) simply as a *pair*. We will define this recursively. Note that a pair (x, H) in a finite-horizon stochastic game is equivalent to a game with stochastic rewards, as all action profiles result in termination of the episode. If all agents play according to private copies of a bandit algorithm $\mathcal{B}$ for T rounds, the average reward for each agent over the period can be viewed as a random variable, where the expected value for agent i is given by:

$$V_i^{\mathcal{B}}(x, H) = \mathop{\mathbb{E}}_{\{a^t\}\sim(\mathcal{B})_{i\in[M]}, \theta\sim r(x,H)}\left[\frac{1}{T}\sum_{t=1}^{T} u_i(a_i^t; a_{-i}, \theta)\right].$$

This will be in $[0, 1]$ for all agents. We can also view other pairs (x, h) as games with stochastic rewards, where the

Preliminary version

immediate reward for an agent is augmented with their value of the state they transition to. Values of states in steps prior to H will represent the expected reward of an agent in the remainder of the episode when all agents play at each state according to $\mathcal{B}$ at each pair, augmenting their immediate payoffs at a pair with the value of the pair they transition to. Suppose $V_i^{\mathcal{B}}(x', h')$ is defined for all $x' \in \mathcal{X}$ and for all $h' > h$. Then,

$$V_i^{\mathcal{B}}(x, h) = \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} u_i(a_i^t; a_{-i}, \theta) + V_i^{\mathcal{B}}(\tau(a, x), h+1)\right],$$

where the expectation is taken over the randomness of each copy of $\mathcal{B}$ as well as sampled reward tensors and transition functions. These will be in $[0, H-h+1]$, but throughout, we will assume that rewards are scaled to $[0, 1]$ before being given to $\mathcal{B}$. These subgame values we have defined represent the utility which an agent can obtain in expectation if they use a copy of $\mathcal{B}$ at each state and know all downstream subgame values. Subgame values can be equivalently defined using downstream value *estimates* $\hat{V}_i$, and we obtain such estimates from Corollary 3.1 which are accurate with high probability.

4.3 AN EFFICIENT OFFLINE ALGORITHM

If we are not constrained to learning online through entire trajectories, and can sample reward tensors and transition functions from any state-step pair (as oracles with constant-time query access), there is a straightforward offline algorithm for computing an EFCE which is a product distribution across pairs.

Algorithm 1: Backward-Inductive Local Learning.

- Use a copy of a bandit algorithm $\mathcal{B}$ for each player to compute an approximate correlated equilibrium and value estimates $\hat{V}_i(x, H)$ for each player and pair, as in Corollary 3.1.
- By backward induction, compute approximate equilibria and value estimates for each pair (x, h) in the same manner, augmenting players' rewards at state x with value estimates for $(x', h+1)$, where x' is the visited state in step $h+1$ for that round.
- Return the product distribution of computed sequences of action profiles across all pairs.

Theorem 4. *BILL computes an ϵ-EFCE in $\tilde{O}(poly(M, N, S, H, 1/\epsilon))$ time.*

The proof is quite similar to the error propagation analysis for Theorem 5.

4.4 PARALLEL LOCAL LEARNING

PLL essentially simulates BILL in a decentralized manner when used by all agents by computing estimates of subgame values for each agent over a series of *epochs*, which are batches of many trajectories of the game, by using a no-swap-regret algorithm $\mathcal{B}$ at each state. We say that a state is *locked* when it is visited enough to obtain an accurate value estimate, and estimates are reset whenever a value estimate for a downstream step is updated. We terminate once an epoch elapses where no new states are locked.

A key point of difficulty here is that visitation probabilities may shift drastically when value estimates change; the sequence of actions taken when all players use $\mathcal{B}$ at a pair may be quite sensitive to small changes in rewards even for just one player. We show that the number of epochs before termination is at most exponential in H, at which point the distribution over action profiles at each pair truncated at the last reset constitutes an approximate correlated equilibrium for the subgame at that pair (given downstream values) almost surely, with the exception of pairs which are visited infrequently under the final value estimates. We then show how regret bounds compose to give an ϵ-EFCE for the entire finite-horizon game when considering action profiles sampled independently across pairs from the aforementioned distributions.

Algorithm 2: Parallel Local Learning. Initialize $\hat{V}_i^{\mathcal{B}}(x, h) = H - h + 1$ for each pair (x, h), as well as a visit counter $c(x, h)$ for each pair set to 0. Let $W = \tilde{\Theta}\left(\frac{S^4 H^7}{\epsilon^2}\right)$ and $L = \Theta\left(\frac{S^2 H^4 W B}{\epsilon^2}\right)$. Initialize a copy of $\mathcal{B}$ at each pair, specified to run for $B = B(\frac{\epsilon}{16H}, N)$ steps. Until termination, run the following procedure for each epoch:

- Run for L trajectories, using $\mathcal{B}$ at each pair, counting rounds and updating actions for a copy of $\mathcal{B}$ only when the corresponding pair is visited. Record rewards as the sum of the observed reward as well as the current value estimate for the *next* pair visited in that trajectory, scaled to [0,1].
- Consider the last step $h \in [H]$ where an unlocked pair's counter crossed $\frac{16H^2WB}{\epsilon}$ in the epoch. Lock all unlocked states at this step with appropriate estimates which were previously unlocked, compute value estimates $\hat{V}_i^{\mathcal{B}}(x, h)$ as the average reward over the corresponding $\frac{16H^2WB}{\epsilon}$ visits, then reset all copies of $\mathcal{B}$, counters, and value estimates at *earlier* pairs $(h' < h)$.
- Terminate if no pair's counter crosses $\frac{16H^2WB}{\epsilon}$ in the epoch.

Note that when all players use this algorithm, locking and unlocking is synchronized across players. The action profile distributions for each pair *after they are last unlocked*

Preliminary version

converge to an approximate EFCE for the game, when we consider action profiles sampled independently for each pair, with a running time at most exponential in the horizon and polynomial in all other parameters.

Theorem 5. *PLL terminates after at most $(S+1)^H + 1$ epochs. After termination, for each pair (x, h), consider the uniform distribution over action profiles $D(x,h)$ played since that pair was last reset. Let D be the distribution over policy profiles where the action profile for each pair (x,h) is sampled independently from $D(x,h)$. With probability at least $1-\delta$, D is an ϵ-EFCE for the game.*

A key step in the analysis of PLL is to bound the number of times that value estimates can change, thus bounding the number of required epochs before estimates stabilize.

Lemma 3. *The algorithm runs for at least H epochs, and at most $(S+1)^H + 1$ epochs.*

Proof. All pairs start unlocked, and some pair in each step is visited at least $\frac{16H^2WB}{\epsilon}$ per epoch by pigeonhole, so the algorithm will not terminate unless there is a locked pair for every step. States are only moved from unlocked to locked at one step per epoch, and so there must be at most H epochs to lock some pair in all steps.

We can bound the number of epochs by bounding the number of times a pair at some step can become locked. Observe that a locked pair at step H will only become locked in one epoch and will never become unlocked afterwards. A pair at step $H-1$ will become locked in at most S epochs, as it will only become locked after at least one pair at step H is locked, and then can be unlocked at most $S-1$ times for the remaining unlocked pairs at step H. In general, the number of epochs in which a state can become locked is bounded by the number of epochs in which a downstream state can become locked. Let $g(h)$ denote this bound on the number of epochs in which a pair at step h can be locked, which is given by:

$$\begin{aligned} g(h) &= \sum_{i=h+1}^{H} Sg(i) \\ &= Sg(h+1) + \sum_{i=h+2}^{H} Sg(i) \\ &= (S+1)g(h+1) \\ &= (S+1)^{H-h} g(H) \\ &= (S+1)^{H-h}, \end{aligned}$$

as $g(H) = 1$. The total number of epochs before termination is then bounded by

$$1 + \sum_{i=1}^{H} Sg(i) = g(0) + 1 = (S+1)^H + 1,$$

accounting for the last epoch in which no states are locked. □

Given this, much of the remainder of the analysis is to analyze the propagation of estimation error and regret terms to give an explicit bound on the regret after estimates have stabilized.

4.5 EFFICIENT LEARNING IN FAST-MIXING STOCHASTIC GAMES

PLL generates an EFCE in polynomial time only when H is a constant. For "fast-mixing" games we give a related algorithm, FastPLL, which converges to an ϵ-EFCE in finite-horizon stochastic games which are γ-*fast-mixing* in time $\tilde{\mathrm{O}}(\mathrm{poly}(S, N, H, 1/\epsilon, 1/\gamma))$. We will say that a finite-horizon stochastic game is γ-fast-mixing if all pairs are visited with probability at least γ in a trajectory when each agent selects a policy uniformly at random, i.e. for each (x,h):

$$\Pr_{\pi\sim\Pi,z}[x \text{ is visited at step } h] \geq \gamma.$$

Unlike the previous algorithm, the fast-mixing assumption allows us to avoid unlocking states once they are locked, as we can ensure sufficient visitation with high probability. As a result, we show that polynomial time convergence to a correlated equilibrium is possible after only H epochs.

Algorithm 3: Fast PLL. Let $B = B\left(\frac{\epsilon}{8H}, N\right)$, and let the epoch length (in trajectories) be given by $L = \tilde{\Theta}\left(\frac{BH^4}{\gamma^{1.5}\epsilon^2}\right)$. Run H epochs, one corresponding to each step (beginning with step H) as follows:

- *Epoch for Step h:* Use a copy of $\mathcal{B}$ to select actions at each pair (x,h), augmenting rewards with computed values for pairs $(x', h+1)$ transitioned to for the next step (if $h < H$). At the end of the epoch, let $\hat{V}_i^{\mathcal{B}}(x,h)$ be the average reward received from all completed runs of $\mathcal{B}$.
- *Upstream ($h' < h$):* Select actions uniformly at random for each pair.
- *Downstream ($h' > h$):* Use $\mathcal{B}$ at each signal as in the epoch for step h', augmenting rewards with value estimates for pairs transitioned to. Restart $\mathcal{B}$ after every B rounds in which it is used, which can include rounds from a prior epoch.

The notion of convergence here is the same as that for PLL.

Theorem 6. *After Algorithm 3 terminates, for each pair (x,h), consider the uniform distribution over action profiles $D(x,h)$ played since epoch $H-h+1$ began. Let D be the distribution over policy profiles where the action profile for each pair (x,h) is sampled independently from $D(x,h)$. With probability at least $1-\delta$, D is an ϵ-EFCE for the game.*

Preliminary Version

5 WHEN CAN WE GET SIMULTANEOUS NO-REGRET?

While PLL gives us a way to generate an EFCE, as well as find stable value estimates for all pairs and players, it is not itself a no-regret algorithm. For single-controller stochastic games, where only one player (the controller) affects transitions, we show that an NFCCE can be reached without shared randomness when the controller uses an algorithm for adversarial MDPs with fixed transitions and each follower uses $\mathcal{B}$ repeatedly in parallel across each pair. Further, both PLL and FastPLL can again be extended to simultaneous no-swap-regret algorithms in the case where *shared randomness* is available for all players.

5.1 EFFICIENT LEARNING IN SINGLE-CONTROLLER STOCHASTIC GAMES

When only one player affects transitions, their problem is equivalent to an adversarial MDP with fixed transitions. The Shifted Bandits U-CO-REPS algorithm from Rosenberg and Mansour [2019] obtains sublinear regret in finite-horizon adversarial MDPs of this form with only bandit feedback and when the transition function is unknown. We show that running Shifted Bandits UC-O-REPS for the "controller" bounds their appropriate notion of regret against arbitrary "followers". The learning problem for the followers can be viewed as a set of *Bayesian* games with shifting signal distributions. In Appendix A.3 we give an extension of our analysis of games with stochastic rewards to Bayesian games, which generalizes the convergence result of Hartline et al. [2015] to remove the "independent private value" assumption, and which we can use to prove a regret bound for a modification of $\mathcal{B}$ (which we call a "parallel bandit" algorithm, denoted $\mathcal{B}_S$) against arbitrary opponents. The regret bound holds even when the "signal distribution" for the Bayesian game shifts over time, and the followers will use a copy of $\mathcal{B}_S$ for each time-step. As such, all agents can efficiently reach an NFCCE by black-box regret minimization.

Algorithm 4: S.B. U-CO-REPS + P.B. Let $B_L(\epsilon)$ be the time after which S.B. U-CO-REPS has per-step regret ϵ, which is poly$(H, S, N, 1/\epsilon)$, and let $B_F(\epsilon) = B(\epsilon/S, N)$. Run for $T = \frac{8\log(M/\delta)}{\epsilon^2} \cdot \max\left(B_L(\epsilon/8), B_F(\epsilon/8)\right)$ total trajectories, where each player acts as follows:

- *Controller:* Select policies for each trajectory using S.B. U-CO-REPS, restarting every $B_L(\epsilon/8)$ trajectories.
- *Followers:* Select policies using a copy of $\mathcal{B}_S$ for Bayesian games at each step, counting only immediate rewards, and restarting every $B_F(\epsilon/8)$ trajectories.

This specifies a policy for each player prior to the start of each trajectory, and this sequence of policies will converge to an approximate NFCCE.

Theorem 7. *With probability at least $1-\delta$, the uniform distribution over the sequence of policy profiles played by Algorithm 4 is an ϵ-NFCCE for the game.*

5.2 SIMULTANEOUS NO-SWAP-REGRET WITH SHARED RANDOMNESS

If players have access to shared randomness at each step, they can play according to the equilibrium generated by PLL or FastPLL in future rounds without any explicit communication. The total regret bound is sublinear in T when the "target average reget" for the PLL (or FastPLL) portion is appropriately calibrated so that the any regret incurred at the beginning does not overwhelm the average regret for the entire sequence of play.

Algorithm 5: PLL with Shared Randomness (PLL-SR). Let $\epsilon_1 = \tilde{\Theta}\left(\sqrt[7]{\frac{N^3 S^{O(H)}}{T}}\right)$ and $\epsilon_2 = \tilde{\Theta}\left(\sqrt[5]{\frac{N^3 H^4 \gamma^{2/3}}{T}}\right)$.

- Run PLL, specified for an ϵ_1-EFCE, until termination, or FastPLL for an ϵ_2-EFCE.
- At each step after termination, each player receives the same uniform random number $w \in [W^*]$ and plays the wth action of the final high-probability local CE sequence (from Corollary 3.1), where W^* is the appropriate length of the sequence for PLL or FastPLL.

Theorem 8. *With respect to $\mathcal{F}$, PLL-SR has regret $\tilde{O}(T^{\frac{6}{7}})$ and FastPLL-SR has regret $\tilde{O}(T^{\frac{4}{5}})$.*

Acknowledgements

We thank Christos Papadimitriou and Tim Roughgarden for helpful feedback and suggestions throughout this work, and Kiran Vodrahalli and Utkarsh Patange for illuminating discussions regarding the hardness result.

Preliminary Version

References

Yasin Abbasi-Yadkori, Peter L Bartlett, Varun Kanade, Yevgeny Seldin, and Csaba Szepesvari. Online learning in markov decision processes with adversarially chosen transition probability distributions. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26, pages 2508–2516. Curran Associates, Inc., 2013.

Robert J. Aumann. Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics*, 1 (1):67–96, 1974. ISSN 0304-4068. doi: https://doi.org/10.1016/0304-4068(74)90037-8.

Dirk Bergemann and Stephen Morris. Bayes correlated equilibrium and the comparison of information structures in games. *Theoretical Economics*, 11(2):487–522, 2016. doi: 10.3982/TE1808.

Avrim Blum and Yishay Mansour. From external to internal regret. volume 8, 05 2004. doi: 10.1007/11503415_42.

Ronen Brafman and Moshe Tennenholtz. R-max - a general polynomial time algorithm for near-optimal reinforcement learning. volume 3, pages 953–958, 01 2001. doi: 10.1162/153244303765208377.

Andrea Celli, Alberto Marchesi, Gabriele Farina, and Nicola Gatti. No-regret learning dynamics for extensive-form correlated and coarse correlated equilibria. *CoRR*, abs/2004.00603, 2020. URL `https://arxiv.org/abs/2004.00603`.

N. Cesa-Bianchi, A. Conconi, and C. Gentile. On the generalization ability of on-line learning algorithms. *IEEE Transactions on Information Theory*, 50(9):2050–2057, 2004. doi: 10.1109/TIT.2004.833339.

H. S. Chang, J. Hu, M. C. Fu, and S. I. Marcus. Adaptive adversarial multi-armed bandit approach to two-person zero-sum markov games. *IEEE Transactions on Automatic Control*, 55(2):463–468, 2010.

Xi Chen and Binghui Peng. Hedging in games: Faster convergence of external and swap regrets. *CoRR*, abs/2006.04953, 2020. URL `https://arxiv.org/abs/2006.04953`.

Xi Chen, Xiaotie Deng, and Shang-Hua Teng. Settling the complexity of computing two-player nash equilibria. *CoRR*, abs/0704.1678, 2007.

Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The complexity of computing a nash equilibrium. In *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '06, page 71–78, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595931341. doi: 10.1145/1132516.1132527.

Eyal Even-Dar, Sham M. Kakade, and Yishay Mansour. Experts in a markov decision process. In *Proceedings of the 17th International Conference on Neural Information Processing Systems*, NIPS'04, page 401–408, Cambridge, MA, USA, 2004. MIT Press.

Gabriele Farina, Tommaso Bianchi, and Tuomas Sandholm. Coarse correlation in extensive-form games. *CoRR*, abs/1908.09893, 2019. URL `http://arxiv.org/abs/1908.09893`.

Jerzy A. Filar and T. E. S. Raghavan. A matrix game solution of the single-controller stochastic game. *Mathematics of Operations Research*, 9(3):356–362, 1984. doi: 10.1287/moor.9.3.356.

Francoise Forges. Five legitimate definitions of correlated equilibrium in games with incomplete information. CORE Discussion Papers 1993009, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), 1993.

Dean P. Foster and Rakesh V. Vohra. Calibrated learning and correlated equilibrium. *Games and Economic Behavior*, 21(1):40–55, 1997. ISSN 0899-8256. doi: https://doi.org/10.1006/game.1997.0595.

Amy Greenwald and Keith Hall. Correlated q-learning. In *ICML*, 2003.

Eric A. Hansen, Daniel S. Bernstein, and Shlomo Zilberstein. Dynamic programming for partially observable stochastic games. In *Proceedings of the 19th National Conference on Artifical Intelligence*, AAAI'04, page 709–715. AAAI Press, 2004. ISBN 0262511835.

Sergiu Hart and Andreu Mas-Colell. A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68:1127–1150, 09 2000. doi: 10.1111/1468-0262.00153.

Jason Hartline, Vasilis Syrgkanis, and Éva Tardos. No-regret learning in bayesian games. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, page 3061–3069, Cambridge, MA, USA, 2015. MIT Press.

Johan Håstad. Some optimal inapproximability results. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, STOC '97, page 1–10, New York, NY, USA, 1997. Association for Computing Machinery. ISBN 0897918886. doi: 10.1145/258533.258536. URL `https://doi.org/10.1145/258533.258536`.

Wan Huang and Bernhard von Stengel. Computing an extensive-form correlated equilibrium in polynomial time. pages 506–513, 12 2008. doi: 10.1007/978-3-540-92185-1_56.

Preliminary version

Michael J. Kearns, Yishay Mansour, and Satinder P. Singh. Fast planning in stochastic games. *UAI*, 2000.

Michael L. Littman. Markov games as a framework for multi-agent reinforcement learning. In *ICML*, 1994.

John F. Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, 1950. ISSN 0027-8424. doi: 10.1073/pnas.36.1.48.

Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, 2007. doi: 10.1017/CBO9780511800481.

Christos H. Papadimitriou and John N. Tsitsiklis. The complexity of markov decision processes. *Mathematics of Operations Research*, 12(3):441–450, 1987. ISSN 0364765X, 15265471.

Aviv Rosenberg and Yishay Mansour. Online stochastic shortest path with bandit feedback and unknown transition function. In *NeurIPS*, 2019.

Yoav Shoham and Kevin Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2008. doi: 10.1017/CBO9780511811654.

Eilon Solan and Nicolas Vieille. Stochastic games. *Proceedings of the National Academy of Sciences*, 112(45): 13743–13746, 2015. ISSN 0027-8424. doi: 10.1073/pnas.1513508112.

Vasilis Syrgkanis, Alekh Agarwal, Haipeng Luo, and Robert E. Schapire. Fast convergence of regularized learning in games. *CoRR*, abs/1507.00407, 2015.

Bernhard von Stengel and Françoise Forges. Extensive-form correlated equilibrium: Definition and computational complexity. *Mathematics of Operations Research*, 33, 11 2008. doi: 10.1287/moor.1080.0340.

K. Zhang, Z. Yang, and T. Basar. Networked multi-agent reinforcement learning in continuous spaces. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 2771–2776, 2018.

Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Basar. Finite-sample analyses for fully decentralized multi-agent reinforcement learning. *CoRR*, abs/1812.02783, 2018.

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

Preliminary version