Two-Timescale Algorithms for Learning Nash Equilibria in General-Sum Stochastic Games

H.L. Prasad[†], Prashanth L.A.[♯] and Shalabh Bhatnagar[♯]

[†]Streamoid Technologies, Inc, INDIA, prasad@streamoid.com
[♯]Indian Institute of Science, INDIA, {prashanth,shalabh}@csa.iisc.ernet.in

ABSTRACT

We consider the problem of finding stationary Nash equilibria (NE) in a finite discounted general-sum stochastic game. We first generalize a non-linear optimization problem from [9] to a general N-player game setting. Next, we break down the optimization problem into simpler sub-problems that ensure there is no Bellman error for a given state and an agent. We then provide a characterization of solution points of these sub-problems that correspond to Nash equilibria of the underlying game and for this purpose, we derive a set of necessary and sufficient SG-SP (Stochastic Game - Sub-Problem) conditions. Using these conditions, we develop two provably convergent algorithms. The first algorithm - OFF-SGSP - is centralized and model-based, i.e., it assumes complete information of the game. The second algorithm - ON-SGSP - is an online model-free algorithm. We establish that both algorithms converge, in self-play, to the equilibria of a certain ordinary differential equation (ODE), whose stable limit points coincide with stationary NE of the underlying general-sum stochastic game. On a single state non-generic game [12] as well as on a synthetic two-player game setup with $810,000$ states, we establish that ON-SGSP consistently outperforms NashQ [16] and FFQ [21] algorithms.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems*; G.1.6 [**Optimization**]: Gradient methods

General Terms

Algorithms, Theory

Keywords

Stochastic games, multi agent reinforcement learning, Nash equilibrium, two timescale stochastic approximation

1. INTRODUCTION

We consider a finite *stochastic game* (also referred to as Markov game (cf. [21])) setting that evolves over discrete time instants. As illustrated in Fig. 1, at each stage and in any given state $x \in \mathcal{X}$, all agents act simultaneously with an action vector $a \in \mathcal{A}(x)$ resulting in a transition to the next state $y \in \mathcal{X}$ according to the transition probability $p(y|x,a)$ as well as a reward vector $\mathbf{r}(x,a)$. No agent gets to know what the other agents' actions are before selecting its own action and the reward $r^i(x,a)$ obtained by any agent i in each stage is dependent of both system state x (common to all agents) and the aggregate action a (which includes other agents' actions). Each individual agent's *sole* objective is maximization of his/her *value function* (expected discounted sum of rewards). However, note that the transition dynamics as well as the rewards depend on the actions of all agents and hence, the dynamics of the game is coupled and not independent. We assume that $\mathbf{r}(x,a)$ and the action vector a picked is made known to all agents after every agent i has picked his/her action a^i, that is, we operate in a *model-free* setting[1]. However, we do not assume that each agent knows other agents' policies, i.e., the distribution from which the actions are picked.

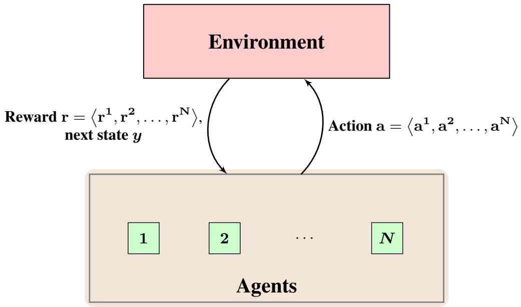

Environment
Reward $\mathbf{r} = \langle \mathbf{r^1}, \mathbf{r^2}, \ldots, \mathbf{r^N} \rangle$,
next state y
Action $\mathbf{a} = \langle \mathbf{a^1}, \mathbf{a^2}, \ldots, \mathbf{a^N} \rangle$
1
2
...
N
Agents

Figure 1: Multi-agent RL setting

The central concept of stability in a stochastic game is that of a *Nash equilibrium.* At a Nash equilibrium point (with a corresponding Nash strategy), each agent plays a *best-response* strategy assuming all the other agents play their equilibrium strategies (see Definition 2 for a precise statement). This notion of equilibrium makes perfect sense in a game setting where agents do not have any incentive to unilaterally deviate from the Nash strategies.

In [7, 9], it is established that finding the stationary NE of a two-player discounted stochastic game is equivalent to solving an optimization problem with a non-linear objective function and linear constraints. We extend this formulation to general N-player stochastic games and observe that this generalization causes the constraints to be non-linear as well. Previous approaches to solving the optimization problem have not been able to guarantee convergence to global minimum, even for the case of $N = 2$. In this light, our contribution is significant as we develop an algorithm to

[1] While the ON-SGSP algorithm that we propose is for this setting, we also propose another algorithm - OFF-SGSP - that is model based.

Appears in: *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015), Bordini, Elkind, Weiss, Yolum (eds.), May 4–8, 2015, Istanbul, Turkey.*

find the global minimum for any $N \geq 2$ via the following steps:
1. First, we break down the main optimization problem into several sub-problems. Each sub-problem can be seen as ensuring no Bellman error, for a particular state $x \in \mathcal{X}$ and agent $i \in \{1, \ldots, N\}$, where $\mathcal{X}$ is the state space and N is the number of agents of the stochastic game considered.
2. Second, we provide a characterization of solution points that correspond to Nash equilibria of the underlying game. As a result, we also derive a set of necessary and sufficient conditions, henceforth referred to as SG-SP (Stochastic Game - Sub-Problem) conditions.
3. Third, using SG-SP conditions, we derive a descent direction that avoids local minima. This is not a steepest descent direction, but a carefully chosen descent direction specific to this optimization problem, which ensures convergence only to points of global minima that correspond to SG-SP points (and hence Nash strategies).
4. Finally, we propose algorithms that incorporate the aforementioned descent direction to ensure convergence to stationary NE of the underlying game.

The algorithms that we propose are as follows:

OFF-SGSP. This is an offline, centralized and model-based scheme, i.e., it assumes that the transition structure of the underlying game is known.

ON-SGSP. This is an online, model-free scheme that is decentralized, i.e., learning is localized to each agent with one instance of ON-SGSP running on each agent. ON-SGSP only requires that other agents' actions and rewards are observed and not their policies, i.e., maps from states to actions.

We make the assumption that for all strategies, the resulting Markov chain is irreducible and positive recurrent. This assumption is common to the analysis of previous multi-agent RL algorithms as well (cf. [15, 21])[2]. To the best of our knowledge, ON-SGSP is the first model-free online algorithm that converges in self-play to stationary NE for any finite discounted general-sum stochastic game where the aforementioned assumption holds.

As suggested in [6], two desirable properties of any multi-agent learning algorithm are as follows:
(a) *Rationality*[3]*:* Learn to play optimally when other agents are stationary; and
(b) *Self-play convergence:* Converge to a Nash equilibrium assuming all agents are using the same learning algorithm.
Our ON-SGSP algorithm can be seen to meet both the properties mentioned above. However, unlike the repeated game setting in [6] (and also in [8]), ON-SGSP solves discounted general-sum stochastic games and possesses theoretical convergence guarantees as well.

The basic idea in both OFF-SGSP and ON-SGSP is to simultaneously update the value and policy tuples for any agent along two timescales (with varying step-sizes) as follows:

Fast timescale. This performs policy evaluation, i.e., estimates the value function using a temporal difference (TD) learning [30] type update rule.

Slow timescale. This performs gradient descent for the policy using a descent direction that ensures convergence to a global minimum of the optimization problem we mentioned earlier.

The formal proof of convergence requires considerable sophistication, as we base our approach on the ordinary differential equations (ODE) method for stochastic approximation [3]. While a few previous papers in the literature have adopted this approach (cf. [2],[32]), their results do not usually start with an algorithm that is shown to track an ODE and instead, an ODE is reached first via analysis and an approximate method is used to solve this ODE. On the other hand, we adopt the former approach and show that both OFF-SGSP and ON-SGSP converge using the following steps:
1. Using two-timescale stochastic approximation, we show that the value and policy updates on the fast and slow timescales, converge respectively to the limit points of a system of ODEs.
2. Next, we provide a simplified representation for the limiting set of the policy ODE and use this to establish that the asymptotically stable limit points of the policy ODE correspond to SG-SP points.
While the first step above uses a well-known result (Hirsch lemma) for analysing stochastic approximation recursions, the techniques used in the second step above are quite different from those used previously. The latter step is crucial in establishing overall convergence, as the strategy π corresponding to each stable limit gives a stationary NE of the underlying general-sum discounted stochastic game.

We demonstrate the practicality of our algorithms on two synthetic two-player setups. The first is a single state non-generic game adopted from [12] that contains two NEs (one pure, the other mixed), while the second is a stick-together game with $810,000$ states (to the best of our knowledge, this cardinality is larger than those considered for general-sum games in previous works). On the first setup, we show that ON-SGSP always converges to NE, while NashQ [16] and FFQ [21] do not in a significant number of experimental runs. On the second setup, we show that ON-SGSP outperforms NashQ and FFQ, while exhibiting a relatively quick convergence rate - requiring approximately 21 iterations per state.

2. RELATED WORK

Various approaches have been proposed in literature for computing Nash equilibrium of general-sum discounted stochastic games and we discuss some of them below.

Multi-agent RL. Littman [20] proposed a minimax Q-learning algorithm for two-player zero-sum stochastic games. Hu and Wellman [15],[16] extended the Q-learning approach to general-sum games, but their algorithms do not possess meaningful convergence guarantees. Friend-or-foe Q-learning (FFQ) [21] is a further improvement based on Q-learning and with guaranteed convergence. However, FFQ converges to Nash only in restricted settings (See conditions A and B in [21]). Moreover, the approaches in [15, 16] require computation of Nash equilibria of a bimatrix game and that in [21] requires solving a linear program, in each round of their algorithms and this is a computationally expensive operation. In contrast, ON-SGSP does not require any such explicit equilibria computation. In [34], the authors show that the traditional Q-learning based approaches are not sufficient to compute Nash equilibria in general-sum games[4].

Policy hill climbing. This is a category of previous works that is closely related to ON-SGSP algorithm that we propose. Important references here include [6], [5], [8] and [33]. All these algorithms are gradient-based, model-free and are proven to converge to NE for stationary opponents in self-play. However, these convergence guarantees are for repeated games only, i.e., the setting is a single state stochastic game, where the objective is to learn the Nash

[2] For the case of stochastic games where there are multiple communicating classes of states or even transient states, a possible workaround is to re-start the game periodically in a random state.

[3] The term *rationality* is not to be confused with its common interpretation in economics parlance.

[4] We avoid this impossibility result by searching for both values and policies instead of just values, in our proposed algorithms.

strategy for a stage-game (see Definition 1 in [8]) that is repeatedly played. On the other hand, we consider general-sum stochastic games where the objective is to learn the best-response strategy against stationary opponents in order to maximize the value function (which is an infinite horizon discounted sum). Further, we work with a more general state space that is not restricted to be a singleton.

Homotopy. In [13] (and also later in [14] and [4]), the authors propose an algorithm, where a homotopic path between equilibrium points of N independent MDPs and the N player stochastic game in question, is traced numerically. This, in turn, gives a Nash equilibrium point of the stochastic game of interest. Their algorithm is closely related to OFF-SGSP in the sense that both are
(i) offline and model-based as they assume complete information (esp. the transition dynamics) about the game; and
(ii) the computational complexity for each iteration of both algorithms grows exponentially with the number of agents N.
(iii) Further, both algorithms are proven to converge to stationary NE, though their approach adopted is vastly different. OFF-SGSP is a gradient descent algorithm designed to converge to the global minimum of a nonlinear program, while the algorithm in [13] involves a tracing procedure to find an equilibrium point.

Linear programming. In [22], the authors solve stochastic games by formulating intermediate optimization problems, called Multi-Objective Linear Programs (MOLPs). However, the solution concept there is *correlated equilibria* and Nash points are a strict subset of this class (and hence are harder to find). Also, the complexity of their algorithm scales exponentially with the problem size.

Both homotopy and linear programming methods proposed in [22] and [13] are tractable only for small sized problems. The computational complexity of these algorithms may render them infeasible on large state spaced games. In contrast, ON-SGSP is a model-free algorithm with a per-iteration complexity that is linear in N, allowing for practical implementations on large state game settings (see Section 9 for one such example with a state space cardinality $= 810,000$). However, per-iteration complexity alone is not sufficient to quantify the performance of an algorithm - see Remark 5 given later.

Rational learning. A popular algorithm with guaranteed convergence to Nash equilibria in general-sum stochastic games is rational learning, proposed by [17]. In their algorithm, each agent i maintains a prior on what he believes to be other agents' strategy and updates it in a Bayesian manner. Combining this with certain assumptions of absolute continuity and grain of truth, the algorithm there is shown to converge to NE. ON-SGSP operates in a similar setting as that in [17], except that we do not assume the knowledge of reward functions. ON-SGSP is a model-free online algorithm and unlike [17], any agent's strategy in ON-SGSP does not depend upon Bayesian estimates of other agents' strategies and hence, their absolute continuity/grain of truth assumptions do not apply.

Evolutionary algorithm. In [1], the authors employ numerical methods in order to solve a system of ODEs and only establish empirical convergence to NE for a group of randomly generated games. In contrast, ON-SGSP is a model-free algorithm that is provably convergent to NE in self-play. We also note that the system of ODEs given in [1] (also found in [32, pp. 189]) turns out to be similar to a portion of the ODEs that are tracked by ON-SGSP.

REMARK 1. *In [26] and [27], the authors question if Nash equilibrium is a useful solution concept for general-sum games. However, if we are willing to concede that* prescriptive, equilibrium agenda *is indeed useful for stochastic games, then we believe our work is theoretically significant. Our ON-SGSP algorithm is a prescriptive, co-operative learning algorithm that observes a sample path from the underlying game and converges to stationary NE. To the best of our knowledge, this is the first algorithm to do so, with proven convergence.*

3. FORMAL DEFINITIONS

A stochastic game can be seen to be an extension of the single-agent Markov decision process. A discounted reward stochastic game is described by a tuple $< N, \mathcal{X}, \mathcal{A}, p, \mathbf{r}, \beta >$, where N represents the number of agents, $\mathcal{X}$ denotes the state space and $\mathcal{A} = \cup_{x \in \mathcal{X}} \mathcal{A}(x)$ is the aggregate action space, where $\mathcal{A}(x) = \prod_{i=1}^{N} \mathcal{A}^i(x)$ is the Cartesian product of action spaces $(\mathcal{A}^i(x))$ of individual agents when the state of the game is $x \in \mathcal{X}$. We assume both state and action spaces to be finite. Let $p(y|x,a)$ denote the probability of going from the current state $x \in \mathcal{X}$ to $y \in \mathcal{X}$ when the vector of actions $a \in \mathcal{A}(x)$ (of the N players) is chosen and let $\mathbf{r}(x,a) = \langle r^i(x,a) : i = 1,2,\ldots,N \rangle$ denote the vector of reward functions of all agents when the state is $x \in \mathcal{X}$ and the vector of actions $a \in \mathcal{A}(x)$ is chosen. Also, $0 < \beta < 1$ denotes the discount factor that controls the influence of the rewards obtained in the future on the agents' strategy (see Definition 1 below).

Notation. $\langle \cdots \rangle$ represents a column vector and $\underline{1}_m$ is a vector of ones with m elements. The various constituents of the stochastic game considered are denoted as follows:
1. **Action:** $a = \langle a^1, a^2, \ldots, a^N \rangle \in \mathcal{A}(x)$ is the aggregate action, a^{-i} is the action of all agents except i and $\mathcal{A}^{-i}(x) := \prod_{j \neq i} \mathcal{A}^j(x)$ is the set of feasible actions in state $x \in \mathcal{X}$ of all agents except i.
2. **Policy:**[5] $\pi^i(x, a^i)$ is the probability of picking action $a^i \in \mathcal{A}^i(x)$ by agent i in state $x \in \mathcal{X}$, $\pi^i(x) = \langle \pi^i(x,a^i) : a^i \in \mathcal{A}^i(x) \rangle$ is the randomized policy vector in state $x \in \mathcal{X}$ for the agent i, $\pi^i = \langle \pi^i(x) : x \in \mathcal{X} \rangle$, $\pi = \langle \pi^i : i = 1,2,\ldots,N \rangle$ is the strategy-tuple and $\pi^{-i} = \langle \pi^j : j = 1,2,\ldots,N, j \neq i \rangle$ is the strategy-tuple of all agents except agent i. We focus only on *stationary strategies* in this paper, as suggested by Theorem 3.
3. **Transition Probability:**
Let $\pi(x,a) = \prod_{i=1}^{N} \pi^i(x,a^i)$ and $\pi^{-i}(x,a^{-i}) = \prod_{j=1, j \neq i}^{N} \pi^j(x,a^j)$. Then, the (Markovian) transition probability from state $x \in \mathcal{X}$ to state $y \in \mathcal{X}$ when each agent i plays according to its randomized strategy π^i can be written as:

$$p(y|x,\pi) = \sum_{a \in \mathcal{A}(x)} p(y|x,a)\pi(x,a).$$

4. **Reward:** $r^i(x,a)$ is the single-stage reward obtained by agent i in state $x \in \mathcal{X}$, where $a \in \mathcal{A}(x)$ is the aggregate action taken.

DEFINITION 1. ***(Value function)*** *The value function is the expected return for any agent $i \in \{1,2,\ldots,N\}$ and is defined as*

$$v^i_\pi(s_0) = E\left[\sum_t \beta^t \sum_{a \in \mathcal{A}(x)} \Big(r^i(s_t,a)\pi(s_t,a) \Big)\right]. \qquad (1)$$

Given the above notion of the value function, the goal of each agent is to find a strategy that achieves a *Nash equilibrium*. The latter is defined as follows:

DEFINITION 2. ***(Nash Equilibrium)*** *A stationary Markov strategy $\pi^* = \langle \pi^{1*}, \pi^{2*}, \ldots, \pi^{N*} \rangle$ is said to be Nash if*

$$v^i_{\pi^*}(s) \geq v^i_{\langle \pi^i, \pi^{-i*} \rangle}(s), \forall \pi^i, \forall i, \forall s \in \mathcal{X}.$$

[5] We use the terms *policy* and *strategy* interchangeably in the paper.

The corresponding equilibrium of the game is said to be Nash equilibrium.

Since we consider a discounted stochastic game with a finite state space, we have the following well-known result that ensures the existence of stationary equilibrium:

THEOREM 3. *Any finite discounted stochastic game has an equilibrium in stationary strategies.*

We shall refer to such stationary randomized strategies as *Nash strategies*. The reader is referred to [10], [31], [28] for a proof of Theorem 3.

4. A GENERALIZED OPTIMIZATION PROBLEM

Basic idea. Using dynamic programming the Nash equilibrium condition in (2) can be written as: $\forall x \in \mathcal{X}, \forall i = 1, 2, \dots, N$,

$$v^i_{\pi^*}(x) = \max_{\pi^i(x) \in \Delta(\mathcal{A}^i(x))} \left\{ E_{\pi^i(x)} Q^i_{\pi^{-i*}}(x, a^i) \right\}, \qquad (2)$$

where

$$Q^i_{\pi^{-i}}(x, a^i) = E_{\pi^{-i}(x)} \left[r^i(x, a) + \beta \sum_{y \in U(x)} p(y|x, a) v^i(y) \right],$$

represents the marginal value associated with picking action $a^i \in \mathcal{A}^i(x)$, in state $x \in \mathcal{X}$ for agent i, while other agents act according to π^{-i}. Also, $\Delta(\mathcal{A}^i(x))$ denotes the set of all possible probability distributions over $\mathcal{A}^i(x)$. The basic idea is to model the objective such that the value function is correct w.r.t. agents' strategies, while add a constraint to ensure that a feasible solution to the problem corresponds to Nash equilibrium.

Objective. A possible optimization objective would be

$$f^i(\mathbf{v}^i, \pi) = \sum_{x \in \mathcal{X}} \left(v^i(x) - E_{\pi^i} Q^i_{\pi^{-i}}(x, a^i) \right),$$

which will have to be minimized over all possible policies $\pi^i \in \Delta(\mathcal{A}^i(x))$. But $Q^i_{\pi^{-i}}(x, a^i)$, by definition, is dependent on strategies of all other agents. So, an isolated minimization of $f^i(\mathbf{v}^i, \pi^i)$ would really not make sense and we consider the aggregate objective $f(\mathbf{v}, \pi) = \sum_{i=1}^{N} f^i(\mathbf{v}^i, \pi)$. This objective which is minimized over all possible policies $\pi^i \in \Delta(\mathcal{A}^i(x))$ of all agents. Thus, we have an optimization problem with objective as $f(\mathbf{v}, \pi)$ along with the natural constraints ensuring that the policy vectors $\pi^i(x)$ remain as probabilities over all possible actions $\mathcal{A}^i(x)$ for all states $x \in \mathcal{X}$ and for all agents $i = 1, \dots, N$.

Constraints. Notice that an optimization problem with the objective discussed above has only a set of simple constraints ensuring that π remains a valid strategy. However, this is not sufficient to accurately represent Nash equilibria of the underlying game. Here, we look at a possible set of additional constraints which might make the optimization problem more useful. Note that the term being maximized in equation (2), i.e., $E_{\pi^i} Q^i_{\pi^{-i}}(x, a^i)$, represents a convex combination of the values of $Q^i_{\pi^{-i}}(x, a^i)$ over all possible actions $a^i \in \mathcal{A}^i(x)$ in a given state $x \in \mathcal{X}$ for a given agent i. Thus, it is implicitly implied that

$$Q^i_{\pi^{-i}}(x, a^i) \leq v^i_{\pi^*}(x), \forall a^i \in \mathcal{A}^i(x), x \in \mathcal{X}, i = 1, 2, \dots, N.$$

Formally, the optimization problem for any $N \geq 2$ is given below:

$$\left.\begin{array}{l}
\min\limits_{\mathbf{v}, \pi} f(\mathbf{v}, \pi) = \sum\limits_{i=1}^{N} \sum\limits_{x \in \mathcal{X}} \left(v^i(x) - E_{\pi^i} Q^i_{\pi^{-i}}(x, a^i) \right) \text{ s.t.} \\
(a)\, \pi^i(x, a^i) \geq 0, \forall a^i \in \mathcal{A}^i(x), x \in \mathcal{X}, i = 1, 2, \dots, N, \\
(b)\, \sum\limits_{i=1}^{N} \pi^i(x, a^i) = 1, \forall x \in \mathcal{X}, i = 1, 2, \dots, N. \\
(c)\, Q^i_{\pi^{-i}}(x, a^i) \leq v^i(x), \forall a^i \in \mathcal{A}^i(x), x \in \mathcal{X}, i = 1, 2, \dots, N.
\end{array}\right\} \qquad (3)$$

In the above, 3(a)–3(b) ensure that π is a valid policy, while 3(c) is necessary for any valid policy to be a NE of the underlying game. Any feasible solution to this problem which makes $f(\mathbf{v}, \pi) = 0$ corresponds to a Nash equilibrium of the underlying game. A proof of this fact for $N = 2$ is in [9, Theorem 3.8.2], while the case of a general $N \geq 2$ can be handled in a similar fashion.

REMARK 2. ***(Difficulty in solving** (3)) In [9], a non-linear optimization problem has been formulated for* zero-sum *stochastic games with two agents ($N = 2$). An associated result (Theorem 3.9.4, page 141, in [9]) states that every local minimum of that optimization problem is also a global minimum. Thus, simple* steepest descent *algorithms are enough to solve for Nash equilibrium strategies in zero-sum stochastic games with two agents. However, this is not the case for general-sum stochastic games, that is, there can be a local minimum which is not a global minimum. This implies that a simple gradient search algorithm which uses the* steepest descent *direction may not converge to Nash equilibrium. In addition to the two-player setting, this remark holds true for general-sum stochastic games with $N \geq 3$ as well.*

4.1 Sub-problems for each state and agent

We form sub-problems from the main optimization problem (3) along the lines of [25], for each state $x \in \mathcal{X}$ and each agent $i \in \{1, 2, \dots, N\}$. The sub-problems are formed with the objective of ensuring that there is no Bellman error (see $g^i_{x,z}(\theta)$ below). For any $x \in \mathcal{X}$, $z = 1, 2, \dots, |\mathcal{A}^i(x)|$ and $i \in \{1, 2, \dots, N\}$, let $\theta := \langle \mathbf{v}^i, \pi^{-i}(x) \rangle$ denote the value-policy tuple and $g^i_{x,z}(\theta) := Q^i_{\pi^{-i}}(x, a^i_z) - v^i(x)$ denote the Bellman error. Further, let $p_z := \pi^i(x, a^i_z)$ and $p = \langle p_z : z = 1, 2, \dots, |\mathcal{A}^i(x)| \rangle$. Then, the sub-problems are formulated as follows:

$$\min_{\theta, p} h_x(\theta, p) = \sum_{z=1}^{|\mathcal{A}^i(x)|} p_z \left[-g^i_{x,z}(\theta) \right] \qquad (4)$$

$$\text{s.t. } g^i_{x,z}(\theta) \leq 0, -p_z \leq 0, \text{ for } z = 1, 2, \dots, |\mathcal{A}^i(x)|,$$
$$\text{and } \sum_z p_z = 1.$$

5. STOCHASTIC GAME - SUB-PROBLEM (SG-SP) CONDITIONS

DEFINITION 4 (SG-SP POINT). *A point $(\mathbf{v}^*, \pi^*)$ of the optimization problem (3) is said to be an SG-SP point if it is a feasible point of (3) and for every sub-problem, i.e., for all $x \in \mathcal{X}$ and $i \in \{1, 2, \dots, N\}$*

$$p^*_z g^i_{x,z}(\theta^*) = 0, \qquad \forall z = 1, 2, \dots, |\mathcal{A}^i(x)|. \qquad (5)$$

The above conditions, which define a point to be an SG-SP point, are called SG-SP conditions. The connection between SG-SP points and Nash equilibria can be seen intuitively as follows:
(i) The objective $f(v^*, \pi^*)$ in (3) can be expressed as a summation of terms of the form $p^*_z[-g^i_{x,z}(\theta^*)]$ over $z = 1, 2, \dots, |\mathcal{A}^i(x)|$ and

over all sub-problems. Condition (5) suggests that each of these terms is zero which implies $f(v^*, \pi^*) = 0$.
(ii) The objective of the sub-problem is to ensure that there is no Bellman error, which in turn implies that the value estimates v^* are correct with respect to the policy π^* of all agents.

THEOREM 5 (NASH $\Leftrightarrow$ SG-SP). *A strategy π^* is Nash if and only if (v^*, π^*) for the corresponding optimization problem (3) is an SG-SP point.*

PROOF.
SG-SP $\Rightarrow$ Nash
The objective function value $f(v^*, \pi^*)$ of the optimization problem (3) can be expressed as a summation of terms of the form $p^*_z[-g^i_{x,z}(\theta^*)]$ over $z = 1, 2, \ldots, m$ and over all sub-problems. Condition (5) suggests that each of these terms is zero which implies $f(v^*, \pi^*) = 0$. From [9, Theorem 3.8.2, page 132], since (v^*, π^*) is a feasible point of (3) and $f(v^*, \pi^*) = 0$, (v^*, π^*) corresponds to Nash equilibrium of the underlying stochastic game.
Nash $\Rightarrow$ SG-SP
From [9, Theorem 3.8.2, page 132], if a strategy π^* is Nash, then a feasible point (v^*, π^*) exists for the corresponding optimization problem (3), where $f(v^*, \pi^*) = 0$. From the constraints of (3), it is clear that for a feasible point, $p^*_z[-g^i_{x,z}(\theta^*)] \geq 0$, for $z = 1, 2, \ldots, m$, for every sub-problem. Since the sum of all these terms, i.e., $f(v^*, \pi^*)$, is zero, each of these terms is zero, i.e., (v^*, π^*) satisfies (5). Thus, (v^*, π^*) is an SG-SP point. □

REMARK 3. *In [25], the authors consider a similar optimization problem as* (3) *for the case of two agents, i.e., $N = 2$ and derive a set of verifiable necessary and sufficient conditions that they call KKT-SP conditions. Our SG-SP conditions are for N-player stochastic games, for any $N \geq 2$. We observe that the simpler and more general (for any N) SG-SP conditions can be used for Nash equilibria as compared to KKT-SP conditions because*
***(i)** every KKT-SP point is also a SG-SP point*[6] *and*
***(ii)** In order to be equivalent to Nash points, KKT-SP conditions require the following additional assumption: For each sub-problem $\left\{\nabla_\theta g^i_{x,z}(\theta^*) : z = 1, 2, \ldots, m\right\}$ is a set of linearly independent vectors. In contrast, SG-SP conditions do not impose any such linear independence requirement to be equivalent to Nash points.*

6. OFF-SGSP: OFFLINE, MODEL-BASED

The basic idea. OFF-SGSP is a two timescale scheme that
(i) estimates the value function $\mathbf{v}$ using value iteration; and
(ii) updates the policy in the descent direction so as to ensure convergence to an SG-SP point.
As mentioned before, OFF-SGSP is a model-based algorithm and the transition dynamics and reward structure of the game are used for both steps above.

Update rule. Along two timescales, OFF-SGSP updates the value-policy tuple (v, π) as follows: For all $x \in \mathcal{X}$ and $a^i \in \mathcal{A}^i(x)$,

$$\pi^i_{n+1}(x, a^i) = \Gamma\left(\pi^i_n(x, a^i) \right. \tag{6}$$
$$\left. - b(n)\sqrt{\pi^i_n(x, a^i)}\left|g^i_{x,a^i}(\mathbf{v}^i_n, \pi^{-i}_n)\right| \overline{\text{sgn}}\left(\frac{\partial f(\mathbf{v}_n, \pi_n)}{\partial \pi^i}\right)\right),$$

$$v^i_{n+1}(x) = v^i_n(x) + c(n)\sum_{a^i \in \mathcal{A}^i(x)} \pi^i_n(x, a^i) g^i_{x,a^i}(\mathbf{v}^i_n, \pi^{-i}_n), \tag{7}$$

where $g^i_{x,a^i}(\mathbf{v}^i, \pi^{-i}) := Q^i_{\pi^{-i}}(x, a^i) - v^i(x)$ denotes the Bellman error, $f(\mathbf{v}, \pi)$ is the objective function in (3) and Γ is a projection operator that ensures that the updates to π stay within the simplex $\mathcal{D} = \{(d_1, \ldots, d_{|\mathcal{A}^i(x)|}) \mid d_i \geq 0, \forall i = 1, \ldots, |\mathcal{A}^i(x)|, \sum\limits_{j=2}^{|\mathcal{A}^i(x)|} d_j \leq 1\}$. $\overline{\text{sgn}}(\cdot)$ is a continuous version of the sign function and projects any x, outside of a very small interval around 0, to ± 1 according to the sign of x (see Remark 6 for a precise definition). Continuity is a technical requirement to ensure convergence. The following assumption on the step-sizes ensures that the π-recursion (6) proceeds on a slower timescale in comparison to the v-recursion (7):

ASSUMPTION 1. *The step-sizes $\{b(n)\}, \{c(n)\}$ satisfy*

$$\sum_{n=1}^{\infty} b(n) = \sum_{n=1}^{\infty} c(n) = \infty, \sum_{n=1}^{\infty}\left(b^2(n) + c^2(n)\right) < \infty, \frac{b(n)}{c(n)} \to 0.$$

Justification for descent direction. The following proposition proves that the decrement for the policy in (6) is a valid descent direction for the objective function $f(\cdot, \cdot)$ in (3).

PROPOSITION 6. *For each $i = 1, 2, \ldots, N$, $x \in \mathcal{X}$, $a^i \in \mathcal{A}^i(x)$, we have that $-\sqrt{\pi^i(x, a^i)}\left|g^i_{x,a^i}(\mathbf{v}^i, \pi^{-i})\right| \overline{\text{sgn}}\left(\dfrac{\partial f(\mathbf{v}, \pi)}{\partial \pi^i}\right)$ is a non-ascent, and in particular a descent direction if $\sqrt{\pi^i(x, a^i)}\left|g^i_{x,a^i}(\mathbf{v}^i, \pi^{-i})\right| \neq 0$, in the objective $f(\mathbf{v}, \pi)$ of (3).*

PROOF. *(Sketch)* The objective $f(\mathbf{v}, \pi)$ can be rewritten as

$$f(\mathbf{v}, \pi) = \sum_{i=1}^{N}\sum_{x \in \mathcal{X}}\sum_{a^i \in \mathcal{A}^i(x)}\left\{\pi^i(x, a^i)\left[-g^i_{x,a^i}(\mathbf{v}^i, \pi^{-i})\right]\right\}.$$

For an $a^i \in \mathcal{A}^i(x)$ for some $x \in \mathcal{X}$ and $i \in \{1, 2, \ldots, N\}$, let

$$\hat{\pi}^i(x, a^i) = \pi^i(x, a^i) - \delta\sqrt{\pi^i(x, a^i)}\left|g^i_{x,a^i}(\mathbf{v}^i, \pi^{-i})\right| \overline{\text{sgn}}\left(\frac{\partial f(\mathbf{v}, \pi)}{\partial \pi^i}\right),$$

for a small $\delta > 0$. Let $\hat{\pi}$ be the same as π except that action a^i is picked as defined above. Then by a Taylor series expansion of $f(\mathbf{v}, \hat{\pi})$ till the first order term, we obtain

$$f(\mathbf{v}, \hat{\pi}) = f(v, \pi) + \delta\left[-\sqrt{\pi^i(x, a^i)}\left|g^i_{x,a^i}(\mathbf{v}^i, \pi^{-i})\right|\right]$$
$$\times \overline{\text{sgn}}\left(\frac{\partial f(\mathbf{v}, \pi)}{\partial \pi^i}\right)\frac{\partial f(\mathbf{v}, \pi)}{\partial \pi^i(x, a^i)} + o(\delta).$$

The rest of the proof amounts to showing that the second term in the expansion above is ≤ 0. This can be inferred as follows:

$$-\sqrt{\pi^i(x, a^i)}\left|g^i_{x,a^i}(\mathbf{v}^i, \pi^{-i})\right| \overline{\text{sgn}}\left(\frac{\partial f(\mathbf{v}, \pi)}{\partial \pi^i}\right)\frac{\partial f(\mathbf{v}, \pi)}{\partial \pi^i}$$
$$= -\sqrt{\pi^i(x, a^i)}\left|g^i_{x,a^i}(\mathbf{v}^i, \pi^{-i})\right|\left|\frac{\partial f(\mathbf{v}, \pi)}{\partial \pi^i}\right| \leq 0,$$

and in particular < 0 if $\sqrt{\pi^i(x, a^i)}\left|g^i_{x,a^i}(\mathbf{v}^i, \pi^{-i})\right| \neq 0$.

Thus, for $a^i \in \mathcal{A}^i(x)$, $x \in \mathcal{X}$ and $i \in \{1, 2, \ldots, N\}$ where $\pi^i(x, a^i) > 0$ and $g^i_{x,a^i}(\mathbf{v}^i, \pi^{-i}) \neq 0$, $f(\mathbf{v}, \hat{\pi}) < f(\mathbf{v}, \pi)$ for small enough δ, while for the remaining cases, $f(\mathbf{v}, \hat{\pi}) \approx f(\mathbf{v}, \pi)$. Note that if $f(\mathbf{v}, \pi) > 0$ which implies that solution is not yet achieved, there is at least one former case. The claim follows. □

7. ON-SGSP: ONLINE AND MODEL-FREE

Though OFF-SGSP is suitable for only off-line learning of Nash strategies, it is amenable for extension to the general (on-line) multi-agent RL setting where neither the transition probability p nor the

[6] Owing to space limitations, we do not provide proof of equivalence between KKT-SP and SG-SP here.

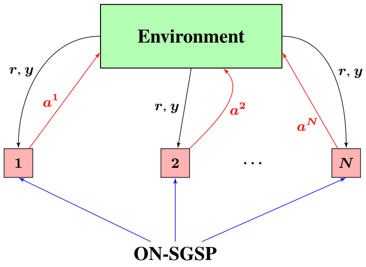


Figure 2: ON-SGSP's decentralized online learning model

reward function r are explicitly known. ON-SGSP operates in the latter model-free setting and uses the stochastic game as a generative model.

As illustrated in Fig. 2, every iteration in ON-SGSP represents a discrete-time instant of transaction with the environment, where each agent presents its action to the environment and observes the next state and the reward vector of all agents. The learning is localized to each agent $i \in \{1, 2, \ldots, N\}$, making the setting decentralized. This is in the spirit of earlier multi-agent RL approaches (cf. [15], [16] and [21]).

Algorithm 1 presents the complete structure of ON-SGSP along with update rules for the value and policy parameters. The algorithm operates along two timescales as follows:

Faster timescale. Each agent estimates its own value function as well as that of other agents, using a temporal-difference (TD) [29] type update in (8). Moreover, the gradient $\dfrac{\partial f(\mathbf{v}_n, \pi_n)}{\partial \pi^i(x, a^i)}$ is also estimated in an online manner via the ξ-recursion in (9). Note that the ξ-recursion is made necessary due to the fact that ON-SGSP operates in a model-free setting.

Slower timescale. The policy update is similar to OFF-SGSP, except that the estimates of value v and gradient ξ are used to derive the decrement in (10).

Note that, since ON-SGSP operates in a model-free setting, both the value and policy updates are different in comparison to OFF-SGSP. The value $\mathbf{v}$ update (8) on the faster timescale can be seen to be the stochastic approximation variant of value iteration and it converges to the same limit as in OFF-SGSP, without knowing the model. On the other hand, the policy update (10) on the slower timescale involves a decrement that is motivated by the descent direction suggested by Proposition 6.

A few remarks about ON-SGSP are in order.

REMARK 4. ***(Coupled dynamics)** In Algorithm 1, an agent i observes the rewards of other agents and uses this information to compute the respective value estimates. These quantities are then used to derive the decrement in the policy update* (10). *This is meaningful in the light of the impossibility result of [11], where the authors show that in order to converge to a Nash equilibrium each agent's strategy needs to factor in the rewards of the other agents.*

REMARK 5. ***(Complexity)** The per-iteration complexity of OFF-SGSP grows exponentially with the number of agents N. Note that the exponential behaviour in N appears because of the computation of expectation over possible next states and strategies of agents. This computation is avoided in ON-SGSP, whose per-iteration complexity is linear in N. In comparison, the stochastic tracing procedure in [13] also has an exponential dependency on N in its per-iteration complexity.*

Algorithm 1 ON-SGSP

```
Input: Starting state $x_0$, initial point $\theta^i_0 = (\mathbf{v}^i_0, \pi^i_0)$, step-sizes $\{b(n), c(n)\}_{n \geq 1}$, number of iterations to run $M >> 0$.
Initialization: $n \leftarrow 1$, $\theta^i \leftarrow \theta^i_0$, $x \leftarrow x_0$
for $n = 1, \ldots, M$ do
    Play action $a^i_n := \pi^i_n(x_n)$ along with other agents in current state $x_n \in \mathcal{X}$
    Obtain next state $y_n \in \mathcal{X}$
    Observe reward vector $r_n =< r^1_n, \ldots, r^N_n >$
    Value Update: For $j = 1, \ldots, N$
    $v^j_{n+1}(x_n) = v^j_n(x_n) + c(n)\left(r^j_n + \beta v^j_n(y_n) - v^j_n(x_n)\right)$    (8)
    Gradient Estimation:
    $\xi^i_{n+1}(x_n, a^i_n) = \xi^i_n(x_n, a^i_n)$    (9)
    $+ c(n)\left(\sum_{j=1}^{N}\left(r^j_n + \beta v^j_n(y_n) - v^j_n(x_n)\right) - \xi^i_n(x_n, a^i_n)\right)$
    Policy Update:
    $\pi^i_{n+1}(x_n, a^i_n) = \Gamma(\pi^i_n(x_n, a^i_n) - b(n)\sqrt{\pi^i_n(x_n, a^i_n)}$
    $\times \left|r^i_n + \beta v^i_n(y_n) - v^i_n(x_n)\right| \overline{\mathrm{sgn}}(-\xi^i_{n+1}(x_n, a^i_n)))$    (10)
end for
```

The per-iteration complexity alone is not sufficient and an analysis of the number of iterations required is necessary to complete the picture[7]. *On the other hand, convergence rate results for general multi-timescale stochastic approximation schemes are not available, see however, [18] for rate results of two timescale schemes with linear recursions.*

8. OUTLINE OF CONVERGENCE PROOF

Owing to space limitations, we provide all the theoretical convergence claims for our algorithms in the following and sketch the proofs for the important ones to explain the main proof ideas. The detailed proof of convergence is available in [24].

Step 1: Analysis of v-recursion

We first show that the updates of $\mathbf{v}$, that are on the faster time-scale, converge to a limit point of the following system of ODEs:$\forall x \in \mathcal{X}, i = 1, 2, \ldots, N$,

$$\frac{dv^i(x)}{dt} = r^i(x, \pi) + \beta \sum_{y \in U(x)} p(y|x, \pi) v^i(y) - v^i(x), \quad (11)$$

where π (that is updated on the slower timescale) is time-invariant.

Let $R^i_\pi = \left\langle r^i(x, \pi), x \in \mathcal{X} \right\rangle$ be a column vector of rewards to agent i and $P_\pi = [p(y|x, \pi), x \in \mathcal{X}, y \in \mathcal{X}]$ be the transition probability matrix, both for a given π.

LEMMA 7. *The system of ODEs (11) has a unique globally asymptotically stable limit point given by*

$$\mathbf{v}^i_\pi = \left[I - \beta P_\pi\right]^{-1} R^i_\pi, i = 1, 2, \ldots, N. \quad (12)$$

THEOREM 8. *For a given π, i.e., with $\pi^i_n \equiv \pi^i$, updates of $\mathbf{v}$ in OFF-SGSP/ON-SGSP satisfy $\mathbf{v}_n \rightarrow \mathbf{v}_\pi$ almost surely, as $n \rightarrow \infty$, where $\mathbf{v}_\pi$ is the globally asymptotically stable equilibrium point of the system of ODEs (11).*

[7] A well-known complexity result [23] establishes that finding the Nash equilibrium of a two-player game is PPAD-complete.

While the above claim is identical for both OFF-SGSP/ON-SGSP, the proofs are quite different. In the former case, it amounts to proving value iteration converges (a standard result in dynamic programming), while the latter case amounts to proving a stochastic approximation variant of value iteration converges (also a standard result in RL).

Step 2: Analysis of π-recursion

Using the converged values $\mathbf{v}_{\pi_n}$ corresponding to the policy π_n on the slower time-scale, we show that updates of π converge to the limit set, say K, of the following system of ODEs:
For all $a^i \in \mathcal{A}^i(x), x \in \mathcal{X}, i = 1, 2, \ldots, N$,

$$\frac{d\pi^i(x,a^i)}{dt} = \bar{\Gamma}\left(\sqrt{\pi^i(x,a^i)}\left|g^i_{x,a^i}(\mathbf{v}^i_\pi, \pi^{-i})\right|\overline{\mathrm{sgn}}(\frac{\partial f(\mathbf{v}_\pi, \pi)}{\partial \pi^i})\right), \tag{13}$$

where $\bar{\Gamma}$ is a projection operator that restricts the evolution of the above ODE to the simplex $\mathcal{D}$.

THEOREM 9. *Assume that the ODE (13) has a compact set K as its set of asymptotically stable equilibrium points. Then, the updates of π, in OFF-SGSP/ON-SGSP, converge to K, almost surely.*

PROOF. *(Sketch)*
OFF-SGSP
The updates of π given by (6) on the slower time-scale $\{b(n)\}$ can be rewritten as: For all $a^i \in \mathcal{A}^i(x), x \in \mathcal{X}$ and $i = 1, 2, \ldots, N$,

$$\pi^i_{n+1}(x,a^i) = \Gamma\left(\pi^i_n(x,a^i) - b(n)\sqrt{\pi^i_n(x,a^i)}\left|g^i_{x,a^i}(v^i_{\pi_n}, \pi^{-i}_n)\right| \times \overline{\mathrm{sgn}}\left(\frac{\partial f(v_{\pi_n}, \pi_n)}{\partial \pi^i}\right) + b(n)\chi_n\right), \tag{14}$$

where χ_n is an $o(1)$ error term that goes to zero as $n \to \infty$. The above recursion can be viewed as an Euler discretization of the system of ODEs (13) with an extra error term χ_n which however (as mentioned before) is $o(1)$. Using a standard stochastic approximation argument, see [19, pp. 191-196], it is easy to see that updates of π in OFF-SGSP converge to the set K.
ON-SGSP[8]
In lieu of the timescale separation between π and v recursions, the fact that Theorem 8 holds for ON-SGSP as well and also the claim regarding ξ-recursion above, one can re-write the update of π in ON-SGSP in a manner similar to (14). However, in the case of ON-SGSP, an additional error term will be introduced owing to the gradient estimating ξ-recursion. This error term is $\hat{\chi}_n = \sqrt{\pi^i_n(x,a^i)}\left[\left|\hat{g}^i_{x,a^i}\right| - \left|g^i_{x,a^i}(v^i_{\pi_n}, \pi^{-i}_n)\right|\right]$. We now claim

$$\left\|\xi^i_n(x,a^i) - \left(-\frac{\partial f(\mathbf{v}_n, \pi_n)}{\partial \pi^i(x,a^i)}\right)\right\| \to 0 \text{ as } n \to \infty \text{ a.s.}$$

In lieu of the above, $\hat{\chi}_n = o(1)$. The rest of the proof follows in a similar manner as OFF-SGSP. □

REMARK 6. *Using the normal sgn() function is problematic for the analysis, as sgn() is discontinuous. We workaround this by employing $\overline{\mathrm{sgn}}()$, which is a continuous extension of sgn(), i.e., it projects any x outside of a small interval around 0 (say $[-\zeta, \zeta]$ for some $\zeta > 0$ small) to either $+1$ or -1 as sgn() would do and within the interval $[-\zeta, \zeta]$, one may choose $\overline{\mathrm{sgn}}(x) = x$ or any other continuous function with compatible end-point values. One could choose ζ arbitrarily close to 0, making $\overline{\mathrm{sgn}}$ practically very close to sgn.*

Since $\overline{\mathrm{sgn}}$ can result in the value 0, one can no longer conclude that $\sqrt{\pi^*}g = 0$ for the points in the equilibrium set K. Note that the former condition (coupled with feasibility) implies it is an SG-SP point. A naive fix would be to change OFF-SGSP/ON-SGSP to repeat an action if $\overline{\mathrm{sgn}}(\cdot)$ returned 0. Henceforth, we shall assume that there are no such $\overline{\mathrm{sgn}}$ induced spurious limit points in the set K.

LEMMA 10. *For all $a^i \in \mathcal{A}^i(x), x \in \mathcal{X}$ and $i = 1, 2, \ldots, N$,*

$$\pi \in K \Rightarrow \pi \in L \text{ and } \sqrt{\pi^i(x,a^i)}g^i_{x,a^i}(v^i_\pi, \pi^{-i}) = 0, \tag{15}$$

where $L = \left\{\pi | \pi(x) \text{ is a probability vector over } \mathcal{A}^i(x), \forall x \in \mathcal{X}\right\}$.

Step 3: Overall convergence to an SG-SP point

Let $G = \left\{\pi \in L \middle| g^i_{x,a^i}(v^i_\pi, \pi^{-i}) \leq 0, \forall a^i \in \mathcal{A}^i(x), x \in \mathcal{X}, i = 1, 2, \ldots, N\right\}$ denote the set of all feasible solutions of (3). The limit set K of (13) can be partitioned using the feasible set G as $K = K_1 \cup K_2$ where $K_1 = K \cap G$ and $K_2 = K \setminus K_1$.

In the following lemma, we show that the set K_2 is the set of locally unstable equilibrium points of (13).

LEMMA 11. *All $\pi^* \in K_2$ are unstable equilibrium points of the system of ODEs (13).*

PROOF. For any $\pi^* \in K_2$, there exists some $a^i \in \mathcal{A}^i(x), x \in \mathcal{X}, i \in \{1, 2, \ldots, N\}$, such that $g^i_{x,a^i}(\mathbf{v}^i_\pi, \pi^{-i}) > 0$ and $\pi^i(x,a^i) = 0$ because K_2 is not in the feasible set G. Let $B_\delta(\pi^*) = \{\pi \in L| \|\pi - \pi^*\| < \delta\}$. Choose $\delta > 0$ such that $g^i_{x,a^i}(\mathbf{v}^i_\pi, \pi^{-i}) > 0$ for all $\pi \in B_\delta(\pi^*) \setminus K$ and consequently $\dfrac{\partial f(\mathbf{v}_\pi, \pi)}{\partial \pi^i} < 0$. So,

$$\bar{\Gamma}\left(-\sqrt{\pi^i(x,a^i)}\left|g^i_{x,a^i}(\mathbf{v}^i_\pi, \pi^{-i})\right|\overline{\mathrm{sgn}}\left(\frac{\partial f(\mathbf{v}_\pi, \pi)}{\partial \pi^i}\right)\right) > 0$$

for any $\pi \in B_\delta(\pi^*) \setminus K$ which suggests that $\pi^i(x,a^i)$ will increase when moving away from π^*. Thus, π^* is an unstable equilibrium point of the system of ODEs (13). □

THEOREM 12. *All asymptotically stable limit points of the system of ODEs (13) correspond to SG-SP (and hence Nash) points of the underlying discounted stochastic game.*

REMARK 7. *Note that from the foregoing, the set K comprises of both stable and unstable attractors and in principle from Lemma 11, the iterates π^i_n governed by (13) can converge to an unstable equilibrium. In our experiments, for every τ iterations, we computed a δ-offset policy $\hat{\pi}^i(x,a^i) = \dfrac{\pi(x,a^i) + \delta}{\sum\limits_{a^i \in \mathcal{A}^i(x)} (\pi(x,a^i) + \delta)}, a^i \in \mathcal{A}^i(x)$. Using the above policy ensures numerically that as $n \to \infty$, $\pi_n \not\to \pi^* \in K_2$, implying convergence of OFF-SGSP/ON-SGSP policies to the stable set K_1. In principle, one may avoid unstable equilibria by including additional noise in the stochastic iterates (see Chapter 4 of [3]). The δ-offset in the policy plays the role of the aforementioned additional noise.*

9. SIMULATION EXPERIMENTS

We test ON-SGSP, NashQ [16] and FFQ [21] algorithms on two general-sum game setups. We implemented Friend Q-learning variant of FFQ, as each iteration of its Foe Q-learning variant involves a computationally intensive operation to solve a linear program.

[8] We let $\{n\}$ denote a subsequence of ON-SGSP iterations when the state is x.

Player 2 → **Player 1** ↓	a_1	a_2	a_3
a_1	$1, 0$	$0, 1$	$1, 0$
a_2	$0, 1$	$1, 0$	$1, 0$
a_3	$0, 1$	$0, 1$	$1, 1$

(a) Payoff matrix.

	NashQ	**FFQ (Friend Q)**	**ON-SGSP**
Oscillate or converge to non-Nash strategy	95%	40%	0%
Converge to $(0.5, 0.5, 0)$	2%	0%	99%
Converge to $(0, 0, 1)$	3%	60%	1%

(b) Results from 100 simulation runs.

Figure 3: Payoff matrix and simulation results for a single state non-generic two-player game

Single State (Non-Generic) Game. This is a simple two-player game adopted from [12], where the payoffs to the individual agents are given in Table 3a. In this game, a strategy that picks a_3 (denoted by $(0, 0, 1)$) constitutes a pure-strategy NE, while a strategy that picks either a_1 or a_2 with equal probability (denoted by $(0.5, 0.5, 0)$) is a mixed-strategy NE.

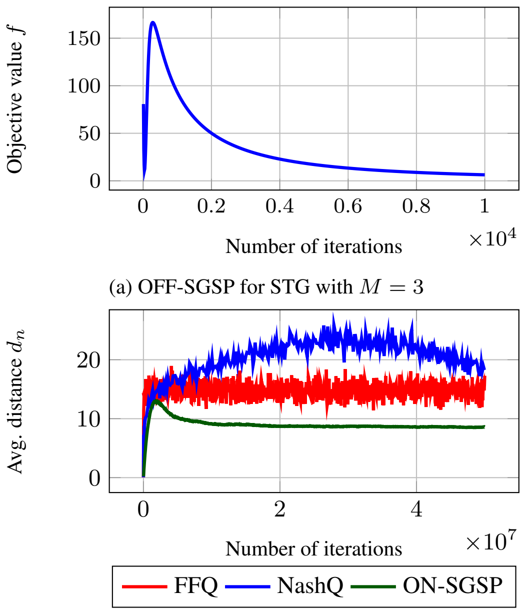
We conduct a stochastic game experiment where each at each stage, the payoffs to the agents are according to Table 3a and the payoffs accumulate with a discount $\beta = 0.8$. We performed 100 experimental runs, with each run corresponding to length of 10000 stages. The aggregated results from this experiment are presented in From Fig. 3b. It is evident that NashQ oscillates and does not converge to NE in most of the runs, while Friend Q-learning converges to a non-Nash strategy tuple in most of the runs. On the other hand, ON-SGSP converges to NE in all the iterations.

Stick Together Game (STG). Here two participating agents located on a rectangular terrain would like to come together and stay close. The state specifies the location of both agents on a rectangular grid of size $M \times M$. The actions available to each agent are to either move to one of the neighbouring cells or stay in the current location. The reward for an agent $i = 1, 2$ for state (x^1, x^2) and action (a^1, a^2) is defined as $r^i(x^i, a^i) = 1 - e^{\|x^1 - x^2\|_1}$. Thus, the reward is zero if the distance between the two agents is zero and a (negative) monotonically decreasing function otherwise. The transition probability function is defined so that it has the highest value towards the next state to which the action points to.

Fig. 4a shows the evolution of the objective function f as a function of the number of iterations for OFF-SGSP. Note that f should go to zero for a Nash equilibrium point.

Fig. 4b shows the evolution of the distance d_n (in ℓ_1 norm) between the agents for a STG game where $M = 30$, which corresponds to $|\mathcal{X}| = 810,000$. Notice that the results are shown only for the model-free algorithms: ON-SGSP, NashQ and FFQ. This is because OFF-SGSP and even the homotopy methods [13] have exponential blow up with M in their computational complexity and hence, are practically infeasible for STG with $M = 30$.

From Fig. 4b, it is evident that following the ON-SGSP strategy, the agents converge to a 4×4-grid within the 30×30-grid. For achieving this result, ON-SGSP takes about 2×10^7 iterations, implying an average $2 \times 10^7/|\mathcal{X}| \approx 21$ iterations per state. While NashQ gets the agents to a 8×8-grid, it is after a large number of iterations ($\approx 5 \times 10^7$). Moreover, from Fig. 4b it is clear that NashQ has not stabilized its strategy in the end. Friend Q-learning gets the agents to 8×8-grid, by driving them to one the corners of the 30×30-grid. While it takes a short number of iterations (≈ 30000) to achieve this, FFQ does not explore the state space well and hence, FFQ's strategy corresponding to the rest of the grid (excluding the corner to which it takes the agents) is not Nash.



(a) OFF-SGSP for STG with $M = 3$

(b) ON-SGSP for STG with $M = 30$

Figure 4: Performance of our algorithms for STG

Runtime performance. We observed that to complete 5×10^7 iterations, ON-SGSP took $\approx$ 42 minutes, while NashQ [16] took nearly 50 hours, as it involves solving for Nash equilibria of a bi-matrix game in each iteration. Friend Q-learning variant of FFQ [21] took $\approx$ 33 minutes (Foe Q-learning variant of FFQ was not implemented owing to its high per-iteration complexity).

10. CONCLUSIONS

In this paper, we derived necessary and sufficient SG-SP conditions to solve a generalized optimization problem and established their equivalence with Nash strategies. We derived a descent (not necessarily steepest) direction that avoids local minima. Incorporating this, we proposed an online model-free algorithm ON-SGSP with guaranteed convergence. Synthetic experiments on two general-sum game setups show that ON-SGSP outperforms two well-known multi-agent RL algorithms.

11. ACKNOWLEDGMENTS

The work was supported by projects from Department of Science and Technology, India and Xerox Corporation, USA.

REFERENCES

[1] N. Akchurina. Multi-agent Reinforcement learning: Algorithm converging to Nash equilibrium in general-sum stochastic games. In *8th International Conference on Autonomous Agents and Multi-agent Systems*, 2009.

[2] N. Akchurina. Multiagent reinforcement learning: algorithm converging to nash equilibrium in general-sum discounted stochastic games. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 725–732. International Foundation for Autonomous Agents and Multiagent Systems, 2009.

[3] V. Borkar. *Stochastic Approximation: A Dynamical Systems Viewpoint*. Cambridge Univ Press, 2008.

[4] R. N. Borkovsky, U. Doraszelski, and Y. Kryukov. A user's guide to solving dynamic stochastic games using the homotopy method. *Operations Research*, 58(4-Part-2):1116–1132, 2010.

[5] M. Bowling. Convergence and no-regret in multiagent learning. *Advances in neural information processing systems*, 17:209–216, 2005.

[6] M. Bowling and M. Veloso. Rational and convergent learning in stochastic games. In *International joint conference on artificial intelligence*, volume 17, pages 1021–1026, 2001.

[7] M. Breton. *Algorithms for stochastic games*. Springer, 1991.

[8] V. Conitzer and T. Sandholm. Awesome: A general multiagent learning algorithm that converges in self-play and learns a best response against stationary opponents. *Machine Learning*, 67(1-2):23–43, 2007.

[9] J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer-Verlag, New York, Inc., 1st edition, November 2004.

[10] A. M. Fink et al. Equilibrium in a stochastic n-person game. *Journal of Science of the Hiroshima University, Series AI (Mathematics)*, 28(1):89–93, 1964.

[11] S. Hart and A. Mas-Colell. Uncoupled dynamics do not lead to Nash equilibrium. *American Economic Review*, pages 1830–1836, 2003.

[12] S. Hart and A. Mas-Colell. Stochastic uncoupled dynamics and nash equilibrium. In *Proceedings of the 10th conference on Theoretical aspects of rationality and knowledge*, pages 52–61. National University of Singapore, 2005.

[13] P. Herings and R. Peeters. Stationary equilibria in stochastic games: Structure, selection, and computation. *Journal of Economic Theory*, 118(1):32–60, 2004.

[14] P. J.-J. Herings and R. Peeters. Homotopy methods to compute equilibria in game theory. Research Memoranda 046, Maastricht : METEOR, Maastricht Research School of Economics of Technology and Organization, 2006.

[15] J. Hu and M. P. Wellman. Multiagent reinforcement learning: Theoretical framework and an algorithm. In *Proceedings of 15th International Conference on Machine Learning*, pages 242–250, 1999.

[16] J. Hu and M. P. Wellman. Nash Q-Learning for general-sum stochastic games. In *Journal of Machine Learning Research*, volume 4, pages 1039–1069, 2003.

[17] E. Kalai and E. Lehrer. Rational learning leads to Nash equilibrium. *Econometrica: Journal of the Econometric Society*, pages 1019–1045, 1993.

[18] V. R. Konda and J. N. Tsitsiklis. Convergence rate of linear two-time-scale stochastic approximation. *Annals of Applied Probability*, pages 796–819, 2004.

[19] H. Kushner and D. Clark. *Stochastic approximation methods for constrained and unconstrained systems*. Springer-Verlag, 1978.

[20] M. L. Littman. Markov games as a framework for multi-agent reinforcement learning. *ICML*, 94:157–163, 1994.

[21] M. L. Littman. Friend-or-Foe Q-Learning in General Sum Games. In *Proceedings of the 18th International Conference on Machine Learning*, pages 322–328. Morgan Kaufmann, 2001.

[22] L. Mac Dermed and C. L. Isbell. Solving stochastic games. *Advances in Neural Information Processing Systems*, 22:1186–1194, 2009.

[23] C. H. Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. *Journal of Computer and System Sciences*, 48(3):498–532, 1994.

[24] H. Prasad, L. Prashanth, and S. Bhatnagar. Two-Timescale Algorithms for Learning Nash Equilibria in General-Sum Stochastic Games. *arXiv preprint arXiv:1401.2086v2*, 2015.

[25] H. L. Prasad and S. Bhatnagar. General-sum stochastic games: verifiability conditions for Nash equilibria. *Automatica*, 48(11):2923–2930, nov 2012.

[26] Y. Shoham, R. Powers, and T. Grenager. Multi-agent reinforcement learning: a critical survey. *Web manuscript*, 2003.

[27] Y. Shoham, R. Powers, and T. Grenager. If multi-agent learning is the answer, what is the question? *Artificial Intelligence*, 171(7):365–377, 2007.

[28] M. J. Sobel et al. Noncooperative stochastic games. *The Annals of Mathematical Statistics*, 42(6):1930–1935, 1971.

[29] R. S. Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44, 1988.

[30] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

[31] M. Takahashi et al. Equilibrium points of stochastic non-cooperative n-person games. *Journal of Science of the Hiroshima University, Series AI (Mathematics)*, 28(1):95–99, 1964.

[32] J. W. Weibull. *Evolutionary Game Theory*. MIT Press, 1996.

[33] C. Zhang and V. R. Lesser. Multi-agent learning with policy prediction. In *AAAI*, 2010.

[34] M. Zinkevich, A. Greenwald, and M. Littman. Cyclic equilibria in Markov games. *Advances in Neural Information Processing Systems*, 18:1641–1648, 2006.