Efficient Learning Equilibrium

Ronen I. Brafman
Computer Science Department
Ben-Gurion University
Beer-Sheva, Israel
e-mail: brafman@cs.bgu.ac.il

Moshe Tennenholtz
Industrial Eng. & Management
Technion
Haifa, Israel 32000
e-mail: moshet@ie.technion.ac.il

March 3, 2003

Abstract

We introduce *efficient learning equilibrium* (ELE), a normative approach to learning in non-cooperative settings. In ELE, the learning algorithms themselves are required to be in equilibrium. In addition, the learning algorithms must arrive at a desired value after polynomial time, and a deviation from the prescribed ELE become irrational after polynomial time. We prove the existence of an ELE (where the desired value is the expected payoff in a Nash equilibrium) and of a Pareto-ELE (where the objective is the maximization of social surplus) in repeated games with perfect monitoring. We also show that an ELE does not always exist in the imperfect monitoring case. Finally, we discuss the extension of these results to general-sum stochastic games.

1 Introduction

Reinforcement learning in the context of multi-agent interaction has attracted the attention of researchers in cognitive psychology, experimental economics, machine learning, artificial intelligence, and related fields for quite some time [13, 6]. Much of this work uses repeated games [5, 8] and stochastic games [16, 15, 12, 2] as models of such interactions.

The literature on learning in games in game theory [8] is mainly concerned with the understanding of learning procedures that **if** adopted by the different agents will converge at the end to an equilibrium of the corresponding game. The game itself may be known; the idea is to show that simple dynamics lead to rational behavior, as prescribed by a Nash equilibrium. The learning algorithms themselves are not required to satisfy any rationality requirement; it is what they converge to, **if** adopted by **all** agents that should be in equilibrium. When facing uncertainty about the game that is played, game-theorists adopt a Bayesian approach. The typical assumption in that approach is that there exists a probability distribution on the possible games, which is common-knowledge. The notion of equilibrium is extended to this context of games with incomplete information, and is treated as the appropriate solution concept. In this context, agents are assumed to be rational agents adopting the corresponding (Bayes-) Nash equilibrium, and learning is not an issue.

Our major claim is that the game-theoretic approach is not in line with the goals of multi-agent reinforcement learning research in AI and must be modified. First, the Bayesian approach used to model partial information is not in line with the common approach in theoretical computer science and computational learning for dealing with uncertainty. Second, the descriptive motivation underlying learning research in game-theory differs considerably from the normative motivation for learning research in AI, and these differences have important ramifications. We now explain these issues in more detail.

First, consider the Bayesian model of partial information. To date, most work in machine learning, and in particular, work on single-agent reinforcement learning has taken a different approach, motivated largely by work on online algorithms in computer science. Here, no distribution is assumed over the uncertain entities, and instead, our goal is to approach the behavior of an agent with complete information as closely and as quickly. Indeed, AI researchers have adopted this non-Bayesian approach in their work on learning in games, looking for algorithms that converge to an appropriate equilibrium in any game out of a class of relevant games; and we follow suit.

However, researches in multi-agent reinforcement learning did choose to adopt other assumptions made by game-theorists, despite the fact that here the differences are much more fundamental. Work on learning in games started with descriptive motivation in mind. That is, its goal was to show that people who use simple heuristic rules for updating their behavior in a multi-agent setting (i.e., in a game) will eventually adopt behavior that corresponds to some appropriate equilibrium behavior. If that is the case, economic models based on equilibria concepts are, in some sense, justified. The assumption that all agents use the same learning rule is justified by the fact that all agents involved are people – i.e., they are all designed similarly. But in AI we are not concerned with descriptive models of human behavior – we are interested in designing artificial agents. Except in the case of cooperative systems, we have no reason to believe that agents designed by different designers will all employ the same learning algorithms. Moreover, one should view the designer's choice of learning algorithm for its agent as a fundamental decision, that should follow normative criteria. Indeed, from the AI perspective, the choice of a learning algorithm is a basic action we take in a game we play against other agent designers.

There is another related point. Game-theorists adopting the descriptive stance are not too concerned with how quickly a learning rule leads to convergence – after all, we had ages to evolve our behavior. But an agent designer wants its agent to learn quickly. He does not care about its agent's "offsprings." Thus, in AI, speed of convergence is of paramount importance.

To better align the research methodology in multiagent reinforcement learning with the AI perspective, we present in this paper a non-Bayesian normative approach to learning in games. Our approach makes no assumptions about the distribution of possible games that may be played – making it more reflective of the setting studied in machine learning and AI and in the spirit of work on on-line algorithms in computer science – and treats the choice of a learning algorithm itself as a game. More specifically, we adopt the framework of repeated games, and view the learning algorithm as a strategy for an agent in a repeated game. This strategy takes an action at each stage based on its previous observations, and initially has no information about the identity of the game being played. Given the above, the following are natural requirements for the learning algorithms provided to the agents:

1. *Individual Rationality:* The learning algorithms themselves should be in equilibrium. It should be irrational for each agent to deviate from its learning algorithm, as long as the other agents stick to their algorithms, *regardless* of what the actual game is.
2. *Efficiency:*
   (a) A deviation from the learning algorithm by a single agent (while the others stick to their algorithms) will become irrational (i.e. will lead to a situation where the deviator's payoff is not improved) after polynomially many stages.
   (b) If all agents stick to their prescribed learning algorithms then the expected payoff obtained by each agent within a polynomial number of steps will be (close to) the value it could have obtained in a Nash equilibrium, had the agents known the game from the outset.

A tuple of learning algorithms satisfying the above properties for a given *class* of games is said to be an *Efficient Learning Equilibrium* (ELE). Notice that the learning algorithms should satisfy the desired properties for *every* game in a given class despite the fact that the actual game played is initially unknown. Such assumptions are typical to work in machine learning. What we borrow from the game theory literature is the criterion for rational behavior in multi-agent systems. That is, we take individual rationality to be associated with the notion of equilibrium. We also take the equilibrium of the actual (initially unknown) game to be our benchmark for success; we wish to obtain a corresponding value although we initially do not know which game is played.

The idea above constitutes the major conceptual contribution of this paper. In the remaining sections we formalize the notion of efficient learning equilibrium, and show that it is not devoid of content, i.e., we prove the existence of an ELE for a general class of games – the class of repeated games with perfect monitoring. We also show that there are classes of games in which an ELE does not exist. Then, we generalize our results to the context of Pareto-ELE (where we wish to obtain maximal social surplus). We also discuss the extension of our results to general-sum stochastic games. Technically speaking, the results we prove rely on a novel combination of the so-called folk theorems in economics, and a novel efficient algorithm for the punishment of deviators in games which are initially unknown.

2 Efficient Learning Equilibrium: Definition

In this section we develop a definition of efficient learning equilibrium in the context of two-player repeated games. The generalization to n-player repeated games is immediate, but requires additional notation, and will not be presented here. An extension to stochastic games appears in Section 6.

A *game* is a model of multi-agent interaction. In a game, we have a set of players, each of whom chooses some action to perform from a given set of actions. As a result of the players' combined choices, some outcome is obtained which is described numerically in the form of a payoff vector, i.e., a vector of values, one for each of the players.

A common description of a (two-player) game is as a bi-matrix. This is called a game in *strategic form*. The rows of the matrix correspond to player 1's actions and the columns correspond to player 2's actions. The entry in row i and column j in the game matrix contains the rewards obtained by the players if player 1 plays his i^{th} action and player 2 plays his j^{th} action. We make the simplifying assumption that the size of the action set of both players is identical. The extension to sets of different sizes is trivial.

In a *repeated game* (RG) the players play a given game G repeatedly. We can view a repeated game, with respect to a game G, as consisting of an infinite number of iterations, at each of which the players have to select an action in the game G. After playing each iteration, the players receive the appropriate payoffs, as dictated by that game's matrix, and move to the next iteration.

For ease of exposition we normalize both players' payoffs in the game G to be non-negative reals between 0 and some positive constant R_{max}. We denote this interval of possible payoffs by $P = [0, R_{max}]$.

In a *perfect monitoring* setting, the set of possible histories of length t is $(A^2 \times P^2)^t$, and the set of possible histories, H, is the union of the sets of possible histories for all $t \geq 0$, where $(A^2 \times P^2)^0$ is the empty history. Namely, the history at time t consists of the history of actions that have been carried out so far, and the corresponding payoffs obtained by the players. Hence, in a perfect monitoring setting, a player can observe the actions selected and the payoffs obtained in the past, but does not know the game matrix to start with. In an *imperfect monitoring* setup, all that a player can observe following the performance of its action is the payoff it obtained and the action selected by the other player. The player cannot observe the other player's payoff. An even more constrained setting is that of *strict imperfect monitoring*, where the player can observe its action and its payoff alone. The definitions of possible histories for an agent in both imperfect monitoring settings follow naturally.

Given an RG, a policy for a player is a mapping from H, the set of possible histories, to the set of possible probability distributions over A. Hence, a policy determines the probability of choosing each particular action for each possible history. Notice that a learning algorithm can be viewed as an instance of a policy.

We define the *value* for player 1 of a policy profile (π, ρ), where π is a policy for player 1 and ρ is a policy for player 2, using the *expected average reward criterion* as follows:

Given an RG M and a natural number T, we denote the expected T-iterations undiscounted average reward of player 1 when the players follow the policy profile (π, ρ), by $U_1(M, \pi, \rho, T)$. The definition for player 2 is similar. We define $U_i(M, \pi, \rho) = \liminf_{T \to \infty} U_i(M, \pi, \rho, T)$ for $i = 1, 2$.

A policy profile (π, ρ) is a *learning equilibrium* if

$$\forall \pi', \rho', U_1(M, \pi', \rho) \leq U_1(M, \pi, \rho), \text{ and } U_2(M, \pi, \rho') \leq U_2(M, \pi, \rho)$$

for every game matrix M (defined over the set A of actions, and the possible payoffs).

Our first requirement, then, is that learning algorithms will be treated as strategies. In order to be individually rational they should be the the best response for one another. In addition, they should rapidly obtain a desired value. The identity of this desired value may be a parameter. We now take a natural candidate, the Nash equilibrium of the game. Another appealing alternative will be discussed later.

Assume we consider games with k actions, $A = \{a_1, \ldots, a_k\}$. For every repeated game M, let $n(G) = (N_1(G), N_2(G))$ be a Nash equilibrium of the (one-shot) game associated with M, and denote by $NV_i(n(G))$ the expected payoff obtained by agent i in that equilibrium.

A policy profile (π, ρ) is an *efficient learning equilibrium*(ELE) if for every $\epsilon > 0, 0 < \delta < 1$, there exists some $T > 0$, polynomial in $\frac{1}{\epsilon}, \frac{1}{\delta}$, and k, such that for every $t \geq T$ and game matrix G (and its corresponding RG, M), $U_i(M, \pi, \rho, t) \geq NV_i(n(G)) - \epsilon$ for $i = 1, 2$, for some Nash equilibrium $n(G)$, and if player 1 deviates from π to π' in iteration l, then $U_1(M, \pi', \rho, l + t) \leq U_1(M, \pi, \rho, l + t) + \epsilon$ with a probability of failure of at most δ. And similarly, for player 2.

Notice that a deviation is considered irrational if it does not increase the expected payoff by more than ϵ. This is in the spirit of ϵ-equilibrium in game theory, and is done in order to cover the case where the expected payoff in a Nash equilibrium equals the probabilistic maximin value.[1] In all other cases, the definition can be replaced by one that requires that a deviation will lead to a decreased value, while obtaining similar results. We have chosen the above in order to remain consistent with the game-theoretic literature on equilibrium in stochastic contexts. Notice also, that for a deviation to be considered irrational, its detrimental effect on the deviating player's average reward should manifest in the near future, not exponentially far in the future.

The above captures the insight of a normative approach to learning in non-cooperative setting. We assume that initially the game is unknown, but the agents will have learning algorithms that will rapidly lead to the values the players would have obtained in a Nash equilibrium had they known the game. Moreover, as mentioned earlier, the learning algorithms themselves should be in equilibrium.

[1] The probabilistic maximin value for player 1 is defined as $\max_\pi \min_\rho U_i(M, \pi, \rho, t)$ where π and ρ range over the set of policies for players 1 and 2, respectively. The definition for player 2 is similar.

3 Efficient Learning Equilibrium: Existence

The definition of ELE is of lesser interest if we cannot provide interesting examples of ELE instances. In this section we prove the following constructive result:

Theorem 1 *There exists an ELE for any perfect monitoring setting.*

Below we describe a concrete algorithm with this property. As we said earlier, it is based on a combination of the so-called folk theorems in economics and a novel, efficient punishment mechanism which ensures the efficiency of our approach. In the folk theorems (e.g., see [9] and the extended discussion in [10]) the basic idea is that any strategy profile that leads to payoffs that are greater than or equal to the security level (probabilistic maximin) values that the agents can guarantee themselves can be obtained by directing the agents to use the prescribed strategies, and telling each agent to punish the other agent if it turns out to deviate from that behavior; the punishment remains a threat that will not be followed in equilibrium and as a result the desired strategy profile will be executed. In order to use this idea in our setting we need however to use a technique for punishing without (initially) knowing the payoff matrix; moreover we need to devise an efficient punishment procedure for that setting.

Recall that we consider a repeated game M, where at each iteration G is played. In what follows, we often use the term *agent* to denote the player using the algorithm in question, and the term *adversary* to denote the other player. Both players have a set $A = \{a_1, \ldots, a_k\}$ of possible actions.

Consider the following algorithm, termed the ELE algorithm.

The ELE algorithm:

Player 1 performs action a_i one time after the other for k times for $i = 1, 2, ..., k$. In parallel to that player 2 performs the sequence of actions $(a_1, \ldots, a_k)$ k times.

If both players behave according to the above, a Nash equilibrium of the (now revealed) game is computed, and the players behave according to the corresponding strategies from that point on. When several Nash equilibria exist, one is selected by using a shared selection algorithm.

If one of the players – whom we refer to as *the adversary* – deviates from the above, the other player – whom we refer to as *the agent*, acts as follows: The agent replaces its payoffs in G by the complements to R_{max} of the adversary payoffs. Hence, the agent will treat the game as a constant-sum game where its aim is to minimize the adversary's payoff. Notice that these payoffs might be unknown. Below we will use G and M to refer to that modified game, and describe how the agent will go about minimizing the adversary's payoff:

Initialize: Construct the following model M' of the repeated game M, where the game G is replaced by a game G' where all the entries in the game matrix are assigned the rewards $(R_{max}, 0)$.

In addition, we associate a boolean valued variable with each joint-action $\{assumed, known\}$. This variable is initialized to the value *assumed*.

Repeat:

Compute and Act: Compute the optimal probabilistic maximin of G' and execute it.

Observe and update: Following each joint action do as follows: Let a be the action the agent performed and let a' be the adversary's action. If (a, a') is performed for the first time, update the reward associated with (a, a') in G', as observed, and mark it *known*.

Claim 1 *The ELE algorithm, when adopted by both players, is indeed an ELE.*

Much of the proof of this theorem, which is non-trivial, rests on showing the agent's ability to punish the adversary quickly. The details are presented in the Appendix.

4 Imperfect monitoring

The ELE algorithm of the previous section uses the agent's ability to view its adversary's actions and payoffs. A natural question is whether this ability is required for the existence of an ELE. In this section we show that in general, perfect monitoring is required, but there are special classes of games in which an ELE exists with imperfect monitoring.

We start with the general case:

Theorem 2 *An ELE does not always exist in the imperfect monitoring setting.*

Proof:

In order to see the above consider the following games:

1. G1:

$$M = \begin{pmatrix} 6, \quad 0 & 0, 100 \\ 5, -100 & 1, 500 \end{pmatrix}$$

2. G2:

$$M = \begin{pmatrix} 6, \;\; 9 & 0, \;\; 1 \\ 5, 11 & 1, 10 \end{pmatrix}$$

The payoffs obtained for a joint action in G1 and G2 are identical for player 1 and are different for player 2.

The only equilibrium of G1 is where both players play the second action, leading to (1,500). The only equilibrium of G2 is where both players play the first action, leading to (6,9). (these are unique equilibria since they are obtained by removal of strictly dominated strategies)

Now, assume that an ELE exists, and look at the corresponding policies of the players in that equilibrium. Notice that in order to have an ELE, we must visit the entry (6,9) most of the times if the game is G2 and visit the entry (1,500) most of the times if the game is G1; otherwise, player 1 (resp. player 2) will not obtain high enough value in G2 (resp. G1), since its other payoffs in G2 (resp. G1) are lower than that.

Given the above, it is rational for player 2 to deviate and pretend that the game is always G1 and behave according to what the suggested equilibrium policy tells it to do in that case. Since the game might be actually G1, and player 1 can not tell the difference, player 2 will be able to lead to playing the second action by both players for most times also when the game is G2, increasing its payoff from 9 to 10, contradicting ELE. ∎

But while our approach is not Bayesian, it does not exclude the possibility that the agent knows that it is participating in a game from a particular class. Thus, there may be classes of repeated games for which an ELE exists. In particular, consider the class of *repeated common-interest games.* These are repeated games M where the underlying game G is a common-interest game, i.e., a game in which both players always receive identical payoffs. In this setting our definition of imperfect and perfect monitoring denote the same setting – if the player knows its payoff, it knows its adversary's payoff as well. Thus, we examine the case of *strict imperfect monitoring*. Recall that in this setting, the player knows only its action and its payoff.

Theorem 3 *There exists ELE for the class of common-interest games under strict imperfect monitoring.*

Proof: The idea is quite simple and, surprisingly, has not been proposed before, while other, more complex and less efficient approaches have been proposed. It does require knowledge of the number of actions available to each agent (or a polynomial bound on them). The algorithm works as follows: First, the agents go through a series of random play. They do so sufficiently many times to ensure that the probability that all joint-actions have been played is greater than $1 - \delta$. During this phase, each agent maintains information about the best payoff obtained so far, and the action it used when this payoff was first obtained. Once the exploration phase is over with, the agent plays this best action repeatedly.

This is a learning equilibria because the average reward this learning strategy leads to is the maximal average reward for every agent. Thus, no agent has any motivation to deviate from it. It is an ELE because a polynomial number of steps is required to attain this average reward (see below) and any deviation will immediately reduce the average reward of the agent. We need only a polynomial number of steps to approximately obtain the maximal average reward because we need only $O(k^4 \cdot log(\frac{k^2}{\delta}))$ steps of random play to ensure that all joint-actions have been played with a probability of at least $1-\delta$. This follows from the following. Given large enough k, we get that the probability that after k^2m trials the agents will not play some previously unplayed joint action can be approximated by e^{-m}. Hence, we get that after $O(k^2 log(\frac{k^2}{\delta}))$ the probability we will not learn the outcome associated with a new joint action can be approximated by $\frac{\delta}{k^2}$. By repeating the process k^2 times we get the desired result. ▮

5 Pareto ELE

The previous sections dealt with ELE in the perfect and imperfect monitoring settings. In both cases we were interested in having a learning procedure that will enable the agents to obtain expected payoffs as the ones they would have obtained in a Nash equilibrium, had they known the game. A more ambitious objective is the following. Let $p_i(a,b)$ denote the payoff for player i in the game in question when player 1 plays a and player 2 plays b. We say that a pair of actions (a,b) is (economically) *efficient*, if $p_1(a,b) + p_2(a,b) = max_{s \in A, t \in A} p_1(s,t) + p_2(s,t)$. That is, if the total payoff for both agents is maximized. It is easy to see that if all the agents can do is to choose an action in G, then there is no general way to guarantee that agents will behave in an economically efficient manner. This is due to the fact that it may be the case that although (a,b) is the only economically efficient behavior, performing a (resp. b) by agent 1 (resp. 2) is irrational: $p_1(a,b)$ (resp. $p_2(a,b)$) may be lower than the probabilistic maximin value that agent 1 (resp. 2) can guarantee itself.

The classical approach in economics for dealing with economic (in)efficiency is by introducing side (monetary) payments. Formally, part of the strategy of agent i is a function $A^2 \times P^2 \rightarrow R_+$, i.e., agent i is instructed to pay a certain amount of money to the other agent as part of its strategy.[2] If an agent's reward is p and he is paid c (where c can be positive, negative, or zero) then his utility is assumed to be $u = p + c$ (this type of utility function is termed quasi linear). The sum, over all agents, of the monetary payments is always 0, and, as a result, if the agents turn out to be using strategies that maximize $u_1 + u_2$ then they will also be economically efficient.

Now we can define the notion of a *Pareto ELE*. A Pareto ELE is similar to a Nash ELE, but its aim is that the agents' behavior will be economically efficient. Therefore, the two distinctive aspects of Pareto ELE are:

1. We require that the agents will be able to get close to an efficient outcome.
2. We allow side payments as part of the agents' behavior.

[2]This is the definition in the perfect monitoring case. The definition in the imperfect monitoring case is similar.

Suppose that we are considering games with k actions. For every repeated game M, let $(P_1(G), P_2(G))$ be an economically efficient joint action of the (one-shot) game associated with M, and denote by $PV_i(M)$ the payoff obtained by agent i in that joint action.

A policy profile (π, ρ), which also allows side payments, is a *Pareto efficient learning equilibrium* if for every $\epsilon > 0, 0 < \delta < 1$, we have that there exists a $T > 0$, polynomial in $\frac{1}{\epsilon}, \frac{1}{\delta}$, and k, such that for every $t \geq T$ and game matrix G (defined over the actions in A), with corresponding RG, M, $U_1(M, \pi, \rho, t) + U_2(M, \pi, \rho, t) \geq PV_1(G) + PV_2(G) - \epsilon$ for $i = 1, 2$, and if player 1 deviates from π to π' in iteration l, then $U_1(M, \pi', \rho, l+t) \leq U_1(M, \pi, \rho, l+t) + \epsilon$ with a probability of failure of at most δ. And similarly for player 2.

Theorem 4 *There exists a Pareto-ELE for any perfect monitoring setting.*

Proof (sketch):

Consider the following algorithm that defines the policies and side payments for the agents.

Player 1 performs k iterations of the action a_i, for $i = 1, 2, ..., k$. In parallel to that, player 2 performs the sequence of actions $(a_1, \ldots, a_k)$ k times.

Now, that the game is known to both agents, they compute the probabilistic maximin values for agent 1 and agent 2. Denote the probabilistic maximin value of agent i by v_i and the payoff it gets in the economically efficient solution by e_i. Without loss of generality, $e_1 - v_1 > e_2 - v_2$. Choose r such that $r = e_1 + e_2 - (v_1 + v_2)$. If player 2 is paid $\frac{r}{2}$ by player 1 when the efficient solution is played then each player's total payoff is at least as high as his probabilistic maximin. This is easy to see by examining the two cases: $e_2 - v_2 > 0$ and $e_2 - v_2 \leq 0$.

From now on the agents adopt the efficient behavior with the above side-payments in all following iterations. If several economically efficient behaviors exist, some predetermined selection algorithm is used.

In case one of players (the adversary) deviates, either in the exploration stage or the following state, the other player (the agent) will punish it as in the case of Nash ELE. It will play as if its payoffs in the game are the the complements to R_{max} of the adversary payoffs.

The proof now follows the steps of the proof for the existence of ELE. ∎

For the case of imperfect monitoring, the same result with respect to Nash-ELE hold here.

Theorem 5 *A Pareto ELE does not always exist in an imperfect monitoring setting.*

6 Stochastic Games

Stochastic games provide a more general model of repeated multi-agent interactions. In a stochastic game the players may be in one of finitely many states, $S = \{s_1, \ldots, s_m\}$, where each state is associated with a game in strategic form. The joint action at each state not only determines the payoffs but also determine (stochastically) the identity of the next state the agents will reach. Formally, let $A = \{a_1, \ldots, a_k\}$ be the set of actions available to the agents. For each state s_i, the game associated with s_i associates a payoff $p^i_j(a, b)$ with agent j when the joint action is (a, b). In addition, for every $s_i \in S$ the probability that the next state will be s_j when the joint action is (a, b) is denoted by $P(s_i, s_j, a, b)$.

Now that we have multiple games, a policy π_i for agent i associates a (possibly mixed) action with every state and, potentially, a payment to the other agent. This policy is a function of the history of states the agent visited and the payoffs it observed. Throughout this section we assume the perfect monitoring setting, since our impossibility result for imperfect monitoring in repeated games immediately rules out the existence of an ELE in the more general context of stochastic games.

Stochastic games provide a more realistic, but also technically more challenging setting. First, let us try and understand the issues involved. The first obstacle we face is the lack of general results on the existence of Nash equilibrium in average-reward stochastic games. Thus, we restrict our attention to the case of Pareto-ELE. Conceptually, the required generalization is straightforward – the learning algorithm should quickly lead to an economically efficient policy for both agents, i.e., a policy that maximizes the average sum of rewards, and deviations should quickly lead to a lower reward. However, while in the case of repeated games we equated "quick" with polynomial in the size of the game and the approximation parameters ϵ and δ, the situation in stochastic games is more complicated. A parameter that is typically used to assess the speed of convergence of a learning algorithm in stochastic games is the ϵ-return mixing time [14, 3]. Intuitively, the ϵ-return mixing time of a policy is the expected time it would take an agent that uses this policy to converge to a value that is ϵ close to the value of the policy. Ideally, we would like a learning algorithm to attain the optimal value in time polynomial in the ϵ-return mixing time of the optimal policy.

Formally, assume some fixed stochastic game M, and let (π_1, π_2) be a policy profile in M. We denote the T-step average reward of this policy profile for agent i starting at state s_0 by $U_i(s_0, \pi_1, \pi_2, T)$, and we define $U_i(s_0, \pi_1, \pi_2) = liminf_{T \to \infty} U_i(s_0, \pi_1, \pi_2, T)$. We let U denote $U_1 + U_2$. The *ϵ-return mixing time* of (π_1, π_2) is the minimal T such that for all $t > T$ and all states s, we have that $U(s, \pi_1, \pi_2, t) > U(s, \pi_1, \pi_2) - \epsilon$. Thus, after the policy profile (π_1, π_2) is executed for T steps or longer, the agents' expected average sum of rewards will be very close to their long-term average sum of rewards. Let (π, ρ) be a policy profile that maximizes $\min_s U(s, \cdot, \cdot)$, and let T_{mix} be its ϵ-return mixing time. The definition of Pareto efficient learning equilibrium in stochastic games is identical to that of repeated games, except that T must be polynomial in T_{mix} as well. Note that if the game is irreducible (i.e., for any fixed policy profile, the induced Markov chain is ergodic), $U(s, \pi, \rho)$ does not depend on s.

We can show the following:

Theorem 6 *Under the following assumptions a Pareto-ELE in stochastic games exists: (1) The agents have perfect monitoring (2) T_{mix} is known.*

Proof: The intuitive idea behind the algorithm is identical to the case of repeated games and so we elaborate only on the new issues. First, the agents run an algorithm for finding a policy profile π, ρ that maximize $U(\cdot, \cdot)$. Next, they run an algorithm for finding the best that each can accomplish on its own (i.e., assuming the other agent is trying to minimize their average payoff). From that point on they run the policy profile π, ρ, adjusted with appropriate side payments so that each agent receives more than the best it can accomplish on its own, much like in the case of repeated games. At any point, if an agent deviates, the other agent plays as if its goal is to minimize the other agent's average reward.

First, note that this learning algorithm is (ϵ-)Pareto-optimal. The long-term average sum of rewards of this algorithm is ϵ-close to the optimal average sum of rewards, as desired. No agent has an incentive to deviate at any stage because the side-payment structure guarantees that it attain at least the value it could attain on its own. To show that this algorithm is a Pareto-ELE, we also need to show that the value can be attained efficiently and punishment can be performed efficiently. We do so by resorting to recent results on efficient learning in fixed-sum stochastic games and common-interest stochastic games. First, to compute the policies π, ρ that maximizes $U(\cdot, \cdot)$ we use the algorithm described in [4]. We refer the reader there for more details. What we need to note here is that this algorithm learns the required policy profile in polynomial time. Next, to compute the values each agent can attain on its own we use R-max [3]. R-max is appropriate here because we are learning in a fixed-sum game. First, we do it for the fixed sum game in which the rewards are based on agent 1's rewards, and then with respect to the fixed-sum game in which the rewards are based on agent 2's rewards. We note that given some value T' as input, R-max will learn a T'-step policy in time polynomial in T' and the other game parameters. This policy will be optimal among all policies that mix in time T'. We shall take $T' = T_{mix}$. The average reward of this policy will be used to compute the side payments structure as in the case of repeated games. In any case, the average reward of the the policy profile π, ρ (suitably modified to include the side payments) will be no lower than the value that each agent can receive on its own. Thus, should an agent deviate from the above, we know that within T_{mix} steps it will attain a lower average reward, i.e., punishment can be carried out efficiently. ▮

Finally, note that there is a standard, though imperfect, technique for removing knowledge of T_{mix} in which we simply guess progressively higher values for T_{mix}. We refer the reader to [3] for the implications of this approach.

7 Discussion

Most previous work on learning in games fits into one of the following two paradigms:

1. The study of learning rules that will lead to a Nash equilibrium (or other solution concept) of a game [8].

2. The study of learning rules that will predict human behavior in non-cooperative interactions, such as the ones modeled in repeated games [6].

While the approach taken in (2) has significant merit for descriptive purposes, a normative approach to learning should go beyond recommending behavior that will eventually lead to some desired solution. The major issues one needs to face are:

1. The learning algorithms of the agents should be individually rational.
2. The learning algorithms should efficiently converge to the desired values if employed by the agents.
3. A deviation from the desired learning algorithm should become irrational after a short period of time.

The concepts introduced in this paper address these issues. Both ELE and Pareto-ELE provide new basic tools for learning in non-cooperative settings. Moreover, we have been able to show constructive existence results for both ELE and Pareto-ELE in the context of repeated games with perfect monitoring. We were also able to show that if we relax the perfect monitoring assumption, the desired properties are impossible to obtain in the general case. Pareto-ELE is an appealing concept in the context of stochastic games as well, and we were able to extend our results to that context. Together, our concepts and results provide a rigorous normative approach to learning in general non-cooperative interactions.

It is useful to contrast our approach with an important line of related work that features algorithms that guarantee that an agent using them will attain a value which is approximately equal to the value he would have attained had he known in advance how his adversary would play. Algorithms along this line appear in e.g., [11] and in [7] (where special attention is given to the issue of efficiency). This latter result is truly in the spirit of on-line algorithms, where our goal is to do as much as we can online as we would have been able to do off-line. In this case, we attempt to react online to an adversary's behavior in a manner that would be similar – in terms of our average payoff – to the best we could have done had we known the adversary's behavior before hand. These results are highly valuable, but many readers may not notice a subtle, but crucial point about them: They treat the adversary's policy as a fixed sequence of mixed strategies (probabilistic actions), and that is contrary to the spirit of game-theory. In reality, the adversary can adjust its policy in response to the agent's behavior. Imagine, for example, the following instance of the well-known Prisoner's Dilemma game:

$$M = \begin{pmatrix} 2, \quad 2 & -10, \ 10 \\ 10, -10 & -5, -5 \end{pmatrix}$$

Consider the following two adversary policies: (1) If the agent initially plays row 1 (denoted *cooperate*) the adversary will always play column 1 (denoted *cooperate*, too). If the agent initially plays row 2 (denoted *defect*) the adversary will always play column 2 (denoted *defect*, too). (2) If the agent initially plays *cooperate* the adversary will always play *defect*. If the agent initially plays *defect* the adversary will always play *cooperate*. It is clear that no agent can guarantee a best-response value against such an adversary, and this approach is limited to a view of the adversary as using a (pre-determined) sequence of mixed strategies. The bottom line is that, despite their practical and theoretical importance, these results cannot replace concepts that are based on the notion of an equilibrium.

Another related work on normative guidelines to the design of learning algorithms is [1]. There, Bowling and Veloso suggest two criteria for learning algorithms. The first, which they call *rationality* stipulates that if the other player's policies converge to a stationary policy then the learning algorithm will converge to the best-response policy. The second, which they call *convergence* stipulates that the learner will necessarily converge to a stationary policy. Both criteria are attractive, but as with the above work, the notion of a Nash-equilibrium of learning strategies is a deeper notion of rationality than that of best-response upon convergence. And convergence, though definitely desirable, ignores the issue of convergence rate. Moreover, convergence, as specified, is not well defined. Indeed, in their work, Bowling and Veloso consider the special (well-defined) case of convergence in self-play, i.e., when all agents use the same algorithm. This is the standard notion of convergence adopted by most work on learning in games uses. In fact, in the particular context of self-play investigated by Bowling and Veloso, their requirements are equivalent to the requirement that the algorithm will converge to a correlated equilibrium – a common property pursued by learning algorithms in the game-theory literature. The concept of ELE provides a more rigorous notion of *individually* rational learning strategies. Moreover, we believe that efficient (i.e., polynomial time) convergence rate should be an integral part of the definition of rationality. In many settings, what happens after an exponential number of iterations is not of great interest. This applies to the judgment of irrationality as well. An agent that makes an "irrational" choice that leads to increased reward in the near future and decreased reward only after an exponential number of steps does not seem too irrational.

References

[1] M. Bowling and M. Veloso. Rational and covergent learning in stochastic games. In *Proc. 17th IJCAI*, pages 1021–1026, 2001.

[2] R. I. Brafman and M. Tennenholtz. R-max – a general polynomial time algorithm for near-optimal reinforcement learning. In *IJCAI'01*, 2001.

[3] R. I. Brafman and M. Tennenholtz. R-max – a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3:213–231, 2002.

[4] R. I. Brafman and M. Tennenholtz. Learning to coordinate efficiently – a model based approach. 2003.

[5] C. Claus and C. Boutilier. The dynamics of reinforcement learning in cooperative multi-agent systems. In *Proc. Workshop on Multi-Agent Learning*, pages 602–608, 1997.

[6] I. Erev and A.E. Roth. Predicting how people play games: Reinforcement learning in games with unique strategy equilibrium. *American Economic Review*, 88:848–881, 1998.

[7] Yoav Freund and Robert E. Schapire. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29:79–103, 1999.

[8] D. Fudenberg and D. Levine. *The theory of learning in games.* MIT Press, 1998.

[9] D. Fudenberg and E. Maskin. The folk theorem in repeated games with discounting or with incomplete information. *Econometrica*, 52:533–554, 1986.

[10] D. Fudenberg and J. Tirole. *Game Theory.* MIT Press, 1991.

[11] S. Hart and A. Mas-Colell. A reinforcement procedure leading to correlated equilibrium. In G. Debreu, W. Neuefeind, and W. Trockel, editors, *Economic Essays: A Festschrift for Werner Hildenbrand*, pages 181–200. Springer, 2001.

[12] J. Hu and M.P. Wellman. Multi-agent reinforcement learning: Theoretical framework and an algorithms. In *Proc. 15th ICML*, 1998.

[13] L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement learning: A survey. *Journal of AI Research*, 4:237–285, 1996.

[14] M. Kearns and S. Singh. Near-optimal reinforcement learning in polynomial time. In *Int. Conf. on Machine Learning*, 1998.

[15] M. L. Littman. Markov games as a framework for multi-agent reinforcement learning. In *Proc. 11th ICML*, pages 157–163, 1994.

[16] L.S. Shapley. Stochastic Games. In *Proc. Nat. Acad. Scie. USA*, volume 39, pages 1095–1100, 1953.

A ELE Existence Proof

Claim 2 *The ELE algorithm, when adopted by both players, is indeed an ELE.*

Proof: We refer to the algorithm used by the agent when playing against the adversary in order to guarantee that the adversary will not obtain more than its maximin value as the *repeated-R-max algorithm*, as it is a version of the R-max algorithm [3].

The proof that the repeated-R-max leads to near-optimal value takes the following steps. The game matrix for G contains k^2 entries. Therefore, the number of iterations in which the agent learns some new information (i.e., has to update its model) is at most k^2. This means that the agent needs to compute its strategy at most $k^2 + 1$ times. Each computation of the probabilistic maximin strategy requires polynomial time and can be carried out using linear programming. As we shall show, except for this small number of phases in which the agent learns new information, its expected payoff is at least as high as the value of the probabilistic maximin strategy. Thus, overall, its average payoff will be high.

The actual number of iterations in which the agent learns new information may be much smaller than k^2, and it depends on how the adversary plays. The adversary can attempt to prevent the agent from learning about various aspects of the game. However, whenever it does this, the agent is guaranteed to obtain a good payoff. More precisely, in any iteration in which the agent did not learn new information, its expected payoff must be no less than the payoff of the probabilistic maximin strategy according to its model, which is at least as high as the probabilistic maximin value of the real game.

Of course, the discussion above deals with expected values. If we wish to guarantee an actual value with probability of at least $1-\delta$, we proceed as follows:

Lemma 1 *Assume that the expected payoff of strategy s is μ. Then, the probability that our average payoff along z iterations will be less than $\mu - \frac{R_{max}}{z^{1/3}}$ is bounded by $e^{-z^{1/3}/2}$.*

Proof: Let X_i be the payoff in iteration i, and let $Y_i = \frac{\mu - X_i}{R_{max}}$. Notice that $|Y_i| \leq 1$, and that $E(Y_i) = 0$. Hence, Chernoff bound implies that: $Prob(\Sigma_{j=1}^{z} Y_j > z^{\frac{2}{3}}) < e^{\frac{-z^{\frac{1}{3}}}{2}}$ This implies that the average return along z iterations is at most $\frac{R_{max}}{z^{\frac{1}{3}}}$ lower than μ with probability of at least $1 - e^{\frac{-z^{\frac{1}{3}}}{2}}$. ∎

In what follows, we choose z so that: $\frac{R_{max}}{z^{1/3}} < \frac{\epsilon}{2R_{max}k^2}$ and $k^2 e^{\frac{-z^{\frac{1}{3}}}{2}} < \theta$. This holds for $z = max(\frac{8R_{max}^6 k^6}{\epsilon^3}, 8ln^3\frac{k^2}{\theta}) + 1$. Note that z is polynomial in k, $\frac{1}{\epsilon}$, $\frac{1}{\theta}$, and R_{max}. The precise value of θ will be determined later, and it will be polynomial in k, $\frac{1}{\epsilon}$, $\frac{1}{\delta}$ and R_{max}, as required.

Assume that the adversary uses some fixed, arbitrary strategy σ, and let us use m to denote the value of the probabilistic maximin strategy for the agent. Our next step is to show the following:

Lemma 2 *Let G' be the agent's current model of the game and let G be the real game. Let ρ be the probabilistic maximin policy with respect to G'. If the agent plays ρ for z iterations then with probability of at least $1-\theta$ the agent will either receive a payoff of at least $m - \frac{\epsilon}{R_{max}}$, or it will learn the value of a new entry.*

Proof: Let r denote the agent's actual average payoff in z iterations of ρ and σ. If $r \geq m - \frac{\epsilon}{R_{max}}$, we are done. Therefore, let us assume that $r < m - \frac{\epsilon}{R_{max}}$. We shall show that with probability of at least $1-\theta$ the agent will learn a new entry in the game matrix.

Let $m_{G'}$ denote the probabilistic maximin value of the game G'. By definition, $m_{G'}$ is also a lower bound on the expected payoff of ρ against σ in G'. Let $\bar{m}$ be the expected value of ρ with respect to G and σ. We know that $|r - \bar{m}| \leq \frac{\epsilon}{2R_{max}}$ with probability of at least $1-\theta$. Since we assume $r < m - \frac{\epsilon}{R_{max}}$, it follows that $\bar{m} < m - \frac{\epsilon}{2R_{max}}$.

However, in G' the payoffs are at least as high as the corresponding values in G. This means that $m_{G'} > m$. Define a new game $W = G' - G$, i.e., the payoff in W is the difference between the corresponding payoffs in G and G'. By definition, W is non-negative. It is strictly positive exactly in the entries that are marked unknown. The expected value of W given ρ and σ is exactly the difference between the expected value of ρ and σ on G' and on G. We know that the expected value of ρ and σ in G' is at least $m_{G'} > m$. The expected value of ρ and σ in G is $\bar{m}$. Therefore, $E(W) > \frac{\epsilon}{2R_{max}}$. Using standard Chernoff bound analysis (e.g., as in Lemma 1, above), it follows that after z iteration, it follows that the actual average value of W is positive with probability of at least 1-θ. This can happen only if an unknown entry is played. ▮

To finish the proof, showing that the agent can punish efficiently a deviator, we choose $Z = zk^2$ and let $T = Z + Q$, where $T\frac{\epsilon}{2} \geq ZR_{max} + Q\frac{\epsilon}{2R_{max}}$. This is satisfied if $Q > Z\frac{2R_{max}}{\epsilon}$. Since Z is polynomial in $k, \frac{1}{\epsilon}, \frac{1}{\theta}$, and R_{max}, we have that Q and T are polynomial in $k, \frac{1}{\epsilon}, \frac{1}{\theta}$, and R_{max}. Finally, we define $\theta = \delta/T$.

We claim that with probability of at least $1-\delta$, for any $t > T$, the average payoff obtained by the agent cannot be lower than the probabilistic maximin value of the game by more than ϵ. We know that with probability of at least $1 - T\theta = 1 - \delta$, in at least Q iterations, the average payoff will be at least $m - \frac{\epsilon}{2R_{max}}$ and in at most Z iteration the average value of the game W will be positive (i.e., learning will occur). We know that the average payoff in these Z remaining iterations is between 0 and R_{max}. Therefore, by definition of T, the overall average value will be as desired.

Finally, we note that the agent need not be aware of any of z, Z, R_{max} or Q, and it simply plays according to the algorithm above.

To conclude the existence proof, we notice that there are only k^2 iterations where the agents explore the unknown game matrix. If a player deviates from the exploration stage, then Lemma 2 guarantees that this deviation will be irrational (up to an ϵ factor, with the desired probability of success). If the agents do follow the exploration stage then deviation from selecting the appropriate strategy (determined by a Nash equilibrium) is irrational by the definition of the Nash equilibrium. ▮