Towards a Pareto-optimal solution in general-sum games

Rajatish Mukherjee & Sandip Sen
Mathematical & Computer Sciences Department
University of Tulsa
rajatish@euler.mcs.utulsa.edu, sandip@kolkata.mcs.utulsa.edu

ABSTRACT

Multi-agent learning literature has looked at iterated two-player games to develop mechanisms that allow agents to learn to converge on Nash Equilibrium strategy profiles. Such equilibrium configuration implies that there is no motivation for one player to change its strategy if the other does not. Often, in general sum games, a higher payoff can be obtained by both players if one chooses not to respond optimally to the other player. By developing mutual trust, agents can avoid iterated best responses that will lead to a lesser payoff Nash Equilibrium. In this paper we consider 1-level agents (modelers) who select actions based on expected utility considering probability distributions over the actions of the opponent(s). We show that in certain situations, such stochastically-greedy agents can perform better (by developing mutually trusting behavior) than those that explicitly attempt to converge to Nash Equilibrium. We also experiment with an interesting action revelation strategy that can give the revealer better payoff on convergence than a non-revealing approach. By revealing, the revealer can convince or encourage other agent to agree to a more trusted equilibrium.

1. INTRODUCTION

Reinforcement learning techniques with performance and convergence guarantees have been developed for isolated single agents. The underlying assumption is that the environment is stationary. Multi-agent or concurrent learning, however, violates this assumption. As a result, the standard reinforcement learning techniques (like Q-learning) are not guaranteed to converge in a multi-agent environment. The desired convergence in multi-agent systems is on an equilibrium strategy-profile (collection of strategies of the agents) rather than optimal strategies for an individual agent.

The stochastic-game (or *Markov Games*) framework, a generalization of Markov Decision Processes for multiple players, has been used to model learning by agents in various domains [2, 3, 4]. In [2], two basic types of multi-agent learners have been studied. The learners who do not model other agents, effectively considering them as passive parts of a non-stationary environment, are called 'independent learners' (ILs). We term these 0-level agents. In contrast to such agents, those that observe others' actions and rewards and use these explicitly in modeling them, are called 'joint-action learners' (JALs). We call these 1-level agents. Theorem 1 in [2] claims that both 0 and 1-level agents converge to equilibria in purely cooperative domains or coordination games. But their work is not extendible to general domains or general-sum games. The authors in [3] have adopted a complete-information general-sum game approach and provide a learning scheme that allows learners to converge to a mixed-strategy Nash Equilibrium in the limit.

Nash Equilibrium, however, does not guarantee that agents will obtain the best possible payoffs, i.e., Nash Equilibrium does not ensure Pareto-optimal solutions. Some non-Nash Equilibrium action combinations may yield better payoffs for both agents, which may be reached if the agents look ahead while selecting actions [1]. Such desirable non-myopic choices are preferred by both agents. While playing best response to other agents' current policy will lead to a deviation from such desirable solutions, restraint or mutual trust can enable players to stick to such action combinations.

In this paper we evaluate the possibility of concurrent learners converging to such desirable non-myopic action choices. While Hu and Wellman's approach is guaranteed to converge to Nash Equilibrium strategy profiles [3] under certain conditions, independent, or even ordinary 1-level Q-learners have no such guarantees. In our previous work, we have observed that 0-level Q-learners often outperformed higher-level Q-learners in the long run even though their learning rate is slower [7]. In this paper we show that greedy modelers can, in their turn, outperform equilibrium seeking modelers in terms of the rewards received. We also investigate an interesting variation of sequential play with action revelation. The motivation behind this work is to determine whether agents can learn to make desirable non-myopic choices by revealing the actions they take to the other agents. By action revelation, we mean that an agent (say A) takes a particular action and communicates its action to the other agent(s). The other agent(s) take their action with full knowledge of agent A's action. We, in the current work, assume that agents are truthful about their action revelation. We design a strategy where each agent is given an opportunity to reveal its action at every alternate iteration (a game consists of multiple iterations) of the game which we refer to as *Alternate revelation choice*. Whether the agent chooses to reveal its action depends on its previous experience (the payoff it received) when it chose to reveal/not reveal its action. Another strategy that we designed and experimented with is the *Simultaneous revelation choice* where both agents are given an opportunity to reveal their actions at every iteration of the game. As in the *Alternate revelation choice*, an agent's choice of action revelation is based on its previous experience (the payoffs received) when it chose to reveal/not reveal its action. We will explain both the strategies in detail later. We present some interesting results in section 4 which seem to indicate that, under certain game matrix configurations (game matrices are discussed in section 2), agents learn to converge to a more desirable Pareto-optimal solution when they learn to reveal their actions. On the contrary, they converge to a myopic Nash equilibrium when they do not adopt the revelation strategies discussed above. However, we still need to formalize our approach and investigate the problem in greater depth before we can determine the matrix configurations under which such results will be obtained. The problem is exacerbated given that the strategies are not guaranteed to converge in a multi-agent environment.

2. LEARNING IN REPEATED GAMES

In this section, we introduce some definitions to formulate a framework for concurrent learning.

Definition 1. *A Markov Decision Process (MDP) is a quadruple $\{S, A, T, R\}$, where S is the set of states, A is the set of actions, T is the transition function, $T : S \times A \rightarrow PD(S)$, PD being a probability distribution, and R is the reward function, $R : S \times A \rightarrow \mathcal{R}$.*

A multi-agent reinforcement-learning task can be looked upon as an extended MDP, with S specifying the joint-state of the agents, A being the joint-actions of the agents, ($A_1 \times A_2 \times \ldots A_n$ where A_i is the set of actions available to the ith agent), T as the joint state-transition function, and the reward function is redefined as $R : S \times A \rightarrow \mathcal{R}^n$. The functions T and R are usually unknown, necessitating learning. The goal of the ith agent is to find a strategy π_i that maximizes its expected sum of discounted rewards,

$$v(s, \pi_i) = \sum_{t=0}^{\infty} \gamma^t E(r_t^i | \pi_i, \pi_{-i}, s_0 = s)$$

where s_0 is the initial joint-state, r_t^i is the reward of the ith agent at time t, $\gamma \in [0, 1)$ is the discount factor, and π_{-i} is the strategy-profile of i's opponents. In [3] the ith agent learns π_{-i} simultaneously, and opts for the best response to it. Though myopically this is the best an agent can do, it may miss opportunities for receiving higher payoffs as in the well-known Prisoner's Dilemma problem.

Definition 2. *A bimatrix game is given by a pair of matrices, (M_1, M_2), (each of size $|A_1| \times |A_2|$) for a two-agent game, where the payoff of the ith agent for the joint action (a_1, a_2) is given by the entry $M_i(a_1, a_2)$, $\forall (a_1, a_2) \in A_1 \times A_2$, $i = 1, 2$.*

Each stage of an extended-MDP for two agents (it can be extended to n agents using n-dimensional tables instead of matrices), can be looked upon as a bimatrix game. In this paper we consider general-sum games where the individual payoffs of the agents for any joint-action are uncorrelated. We now define Nash equilibrium for such games.

Definition 3. *A pure-strategy Nash Equilibrium for a bimatrix game (M_1, M_2) is a pair of actions (a_1^*, a_2^*) such that*

$$M_1(a_1^*, a_2^*) \geq M_1(a_1, a_2^*) \quad \forall a_1 \in A_1$$

$$M_2(a_1^*, a_2^*) \geq M_2(a_1^*, a_2) \quad \forall a_2 \in A_2$$

In a Nash equilibrium the action chosen by each player is the best response to the opponent's current strategy and no player in this game has any incentive for unilateral deviation from its current strategy. A general-sum bimatrix game may not have any pure-strategy Nash Equilibrium.

Definition 4. *A mixed-strategy Nash Equilibrium for a bimatrix game (M_1, M_2) is a pair of probability vectors (π_1^*, π_2^*) such that*

$$\pi_1^*{}' M_1 \pi_2^* \geq \pi_1{}' M_1 \pi_2^* \quad \forall \pi_1 \in PD(A_1)$$

$$\pi_1^*{}' M_2 \pi_2^* \geq \pi_1^*{}' M_2 \pi_2 \quad \forall \pi_2 \in PD(A_2)$$

where $PD(A_i)$ is the set of probability-distributions over the action space of the ith agent.

A significant property of mixed-strategy Nash Equilibria, is that there always exists at least one such equilibrium profile for an arbitrary finite bimatrix game [8]. Given such a bimatrix game (M_1, M_2), the mixed-strategy Nash Equilibrium, (π_1^*, π_2^*), can be computed using a quadratic programming approach as outlined in [6].

We are interested in a non-myopic equilibrium where a player not only considers its best response to current playing trends, but also future possible retaliation by the other player. For example, consider the two players playing π_1^A and π_1^B respectively and the first player getting $\pi_1^A M_A \pi_1^B$ as a result. While considering another strategy π_2^A, A now considers not only if $\pi_2^A M_A \pi_1^B > \pi_1^A M_A \pi_1^B$, but also if $\pi_2^A M_A \pi_1^B > \pi_2^A M_A \pi_2^B$, where π_2^B is Bs best response to π_2^A (this equilibrium concept is similar in motivation to the non-myopic equilibrium in the Theory of Moves approach [1]). Of course, it is difficult to estimate the other player's best response, but this can be approximated based on past play of the opponent.

3. Q-LEARNING

A general, single-agent reinforcement learning task is an MDP, where the state transition and reward functions T and R are unknown. A simple, model-free and on-line technique for reinforcement learning is Q-learning [11]. In a stateless domain, as is the case with single-stage games studied in this paper, an independent Q-learner will have Q-values for each action a, $Q(a)$, and update them based on rewards r received from taking action a as follows:

$$Q(a) \leftarrow Q(a) + \alpha(r - Q(a))$$

where α is the learning-rate. This iteration has been proved to converge to optimal Q-values, for a particular structure of α, but independent of any particular exploration strategy provided it satisfies some general requirements. When a number of independent learners apply this algorithm, the convergence-guarantee does not hold due to the non-stationarity of the environment. However, such straightforward applications of Q-learning in multi-agent systems have achieved success in the past [2, 9, 10, 12]. Our 1-level Q-learners learn Q-values, $Q(a,b)$, for each possible joint-action (a,b), using its observation of the actions of the other agents, but solely its own reward for joint-action. Thus the updation-rule used is

$$Q(a,b) \leftarrow Q(a,b) + \alpha(r - Q(a,b))$$

To allow these 1-level Q-learning agents to increasingly exploit their learned strategies, we use the Boltzmann exploration strategy, which slowly increases the exploitation probability. In this exploration scheme, the action a is selected with probability

$$\frac{e^{E(Q(a))/T}}{\sum_{a'} e^{E(Q(a'))/T}},$$

where $E(a) = \sum_b p_b Q(a,b)$, p_b being computed as the relative-frequency measure from B's action history. Thus we call these agents "expected utility based probabilistic learners" or (EUPs). The temperature parameter T is started at a high value (causing more exploration) and then decreased over time, e.g., by multiplying with a decay factor, to increase the exploitation probability.

We have also experimented with an interesting variation of sequential play with action revelation. We allow one player to reveal or announce its move at each iteration of the game. The other player can choose its move based on complete knowledge of the move made by its opponent. It might still decide to explore its actions instead of playing best response in order to thoroughly evaluate its options. actions. In the revealing version of the game, the players keep not only an estimate of p_b, the frequency distribution of its opponent's moves, but also the corresponding conditional frequency distribution, $p_{b|a}$, i.e., the likelihood that the opponent is going to play its move b if the revealer plays b. Let us consider that each agent has a set of n actions to choose from. The EUPs have to keep an estimate of each of the n actions. However, in the revealing scenario, each agent can reveal any of its n actions or may choose not to reveal its action. So, for each agent, we have to keep an estimate of $2n$ actions (an estimate of an action when it reveals it and an estimate of the same action when it does not reveal it). In the following discussion, a_r refers to a revealed action and a_{nr} refers to a non-revealed action. The Q matrix has entries for all action pairs $Q(i,j)$ where $i \in [1,n]$ and $j \in [1,n]$. Also, a_r and a_{nr} can take values between 1 and n (including 1 and n). Formally, in the exploration scheme, any action a belonging to the set of non-revealed actions is selected with probability

$$\frac{e^{E(Q(a))/T}}{\sum_{a_{nr}} e^{E(Q(a_{nr}))/T} + \sum_{a_r} e^{E'(Q(a_r))/T}},$$

where $E(a_{nr}) = \sum_b p_b Q(a_{nr},b)$ and $E'(a_r) = \sum_b p_{b|a_r} Q(a_r,b)$ and any action a' belonging to the set of revealed actions is selected with probability

$$\frac{e^{E'(Q(a'))/T}}{\sum_{a_{nr}} e^{E(Q(a_{nr}))/T} + \sum_{a_r} e^{E'(Q(a_r))/T}}.$$

Note that a and a' can take any value between 1 and n (including 1 and n).

We explored two variations of the revelation strategy.

- **Alternate revelation choice:** In this strategy, each agent is given an opportunity to reveal its action at every alternate iteration of the game (Any game has 1000 iterations in all our experiments).

- **Simultaneous revelation choice:** In this strategy, both the agents are given an opportunity to reveal their actions at every iteration of the game. If both players agree on revealing, we randomly (with equal probability) choose between the two players. Otherwise, the player who learns to reveal is allowed to do so, and the other player chooses its action based on complete knowledge of the move made by its opponent. The primary difference between the two strategies is that *Simultaneous revelation choice* determines the revealer at every iteration of the game whereas *Alternate revelation choice* has a predetermined revealer and determines whether this agent wants to reveal its action or not. The advantage of *Simultaneous revelation choice* over *Alternate revelation choice* is as follows: Supposing one agent (A) learns to reveal its action, whereas the other (B) does not. Also, when A reveals its action, payoff for both A and B is higher than when B does not reveal its action (otherwise A will have no incentive to reveal its action). Given this, in *Alternate revelation choice*, approximately 50% of the time, B will be given the opportunity to reveal its action (given that B has learnt not to reveal, it will not use the opportunity) whereas in *Simultaneous revelation choice*, A will always get the opportunity to reveal its action (since B will refrain from revealing) and thus, the average payoff for both agents will be higher in *Simultaneous revelation choice*.

4. EXPERIMENTS

Our experimental work uses four game matrices (figure 1, 3, 5 and 7) to highlight how the agents learn to increase their individual rewards by revealing their actions. We experiment with 3×3 game matrices. Each agent has three actions to choose from, where a_is are the actions of agent A and b_is those of agent B. For any action combination, the top-right value in the corresponding matrix cell is the payoff to agent B and the bottom-left value is the payoff to agent A. The shaded entry in each matrix corresponds to the Nash Equilibrium strategy-profile. The action-profile that the agents prefer (greedy) and the desirable non-myopic solutions are also marked in each game-matrix. Our experiments are designed to evaluate the EUPs with no revelation, EUPs with Alternate revelation choice and EUPs with Simultaneous revelation choice.

4.1 Choice of Matrices

We use the four matrices to demonstrate the following results:

- **Matrix 1** (see figure 1) is used to demonstrate how the two agents learn to choose the Nash Equilibrium and not the Pareto-optimal solution irrespective of the strategy chosen.

- **Matrix 2** (see figure 3) is used to demonstrate how the two agents learn to choose the desirable Nash Equilibrium (which incidentally is the Pareto-optimal solution) irrespective of the strategy chosen.

- **Matrix 3** (see figure 5) is used to demonstrate how the agents learn to choose the desirable Nash Equilibrium (which incidentally is the Pareto-optimal solution) using Alternate revelation choice and Simultaneous revelation choice whereas EUPs fail to reach the desired solution.
- **Matrix 4** (see figure 7) is used to demonstrate how Simultaneous revelation choice outperforms Alternate revelation choice which, in turn, outperforms EUPs.

4.2 Experiments with Matrix 1

The matrix in figure 1 has a single pure Nash Equilibrium given by the action-profile $\langle a_3, b_3 \rangle$ giving a payoff of 5 to both agents. The desirable solution (Pareto-optimal), however, is for the action-combination $\langle a_1, b_1 \rangle$ giving a payoff of 10 to both agents. We used two EUPs using the above Q-learning algorithm, learning for 1000 iterations and using 0.99 as the temperature decay factor starting at $T = 10$. The probabilities of adopting joint-actions $\langle a_1, b_1 \rangle$ and $\langle a_3, b_3 \rangle$ as measured by frequencies were recorded every 100 interactions averaged over the last 100 interactions. The values in the figures were averaged over 10 runs, and these probabilities are plotted in figure 2 (left). In this case, the EUPs converge to the Nash Equilibrium in most of the runs even though the payoff is less than the desirable payoff. This is because the payoff matrix is constructed such that a_3 is the best response (actually in this example, a_3 and b_3 are also the agents' dominant strategies) of agent A irrespective of B's choice and b_3 is the best response of agent B irrespective of A's choice.

We achieved similar results when we incorporated Alternate revelation strategy and Simultaneous revelation strategy in our agents. The probabilities of adopting joint-actions $\langle a_1, b_1 \rangle$ and $\langle a_3, b_3 \rangle$ are plotted in figure 2 (middle and right).

4.3 Experiments with Matrix 2

The matrix in figure 3 has both $\langle a_1, b_1 \rangle$ and $\langle a_3, b_3 \rangle$ as pure Nash Equilibria. $\langle a_1, b_1 \rangle$ is also the Pareto-optimal solution. The EUPs learn to adopt the desirable action combination $\langle a_1, b_1 \rangle$ in most runs as shown in the probability plot in figure 4 (left). A similar result is obtained in both Alternate and Simultaneous revelation. The probability plots are shown in figure 4 (middle and right).

4.4 Experiments with Matrix 3

The matrix in figure 5 has two pure Nash Equilibria given by the action-profile $\langle a_3, b_3 \rangle$ giving a payoff of 5 to both agents and the action-profile $\langle a_1, b_1 \rangle$ giving a payoff of 10 to both agents. The desirable solution, however, is for the action-combination $\langle a_1, b_1 \rangle$ giving a payoff of 10 to both agents. In this case, the EUPs converge to the undesirable Nash Equilibrium in most of the runs even though the payoff is less than the desirable payoff. This is because the payoff matrix is constructed such that a_3 is the best response (actually in this example, average payoffs for actions a_3 and b_3 are higher than actions a_1 and b_1) of agent A irrespective of B's choice and b_3 is the best response of agent B irrespective of A's choice. The probabilities of adopting joint-actions $\langle a_1, b_1 \rangle$ and $\langle a_3, b_3 \rangle$ are plotted in figure 6 (left).

The quadratic programming approach [3] produced a mixed strategy (probability distribution) of $[0, 0, 1]$ and $[0, 0, 1]$ for the agents A and B respectively. This corresponds to selecting the $\langle a_3, b_3 \rangle$ action combination. Thus, our EUPs learn almost the same strategy as the mixed-strategy learners seeking Nash Equilibrium.

In both Alternate and Simultaneous revelation scheme, the agents learn that their best response is to select action 1 when the other agent selects action 1 as shown in figure 6 (middle and right). When agent A reveals action 1, agent B (see figure 5) will have higher probability of choosing action 1 and vice versa.

4.5 Experiments with Matrix 4

In the game matrix in figure 7, $\langle a_3, b_3 \rangle$ is the only pure Nash Equilibrium. However, $\langle a_1, b_1 \rangle$ is the desirable solution. From figure 8 (left) we can see that the EUPs learn to select $\langle a_3, b_3 \rangle$ (the Nash Equilibrium solution).

The profile learned by 1-level mixed strategy agent for the matrix in figure 7 (left) is $[0.09, 0, 0.91]$ and $[0.09, 0, 0.91]$ for A and B respectively. This gives an expected reward of 5.45 to each of the mixed-strategy equilibrium learners, whereas our EUPs receive expected reward of 5.0 for selection of the joint-action $\langle a_1, b_1 \rangle$ alone.

In the Alternate revelation scheme/strategy, the agents take actions $\langle a_1, b_1 \rangle$ and $\langle a_3, b_3 \rangle$ with almost equal probability (see figure 8 (middle)). Thus, the expected reward for the agents is more when they reveal their action than when they do not do so (EUPs).

Finally, in the Simultaneous revelation scheme/strategy, the agents choose the action-profile $\langle a_1, b_1 \rangle$ in most of the runs (see figure 8 (right)). Thus, the agents learn to choose the desirable action-pair combination in this scheme/strategy.

In the Alternate revelation scheme, each agent is given a chance to reveal irrespective of whether it has learnt to reveal or not. In the above experiments using revelation schemes, A learns not to reveal its action (whenever A reveals action 1, B exploits A by taking action 3) whereas B has learnt to reveal its action (action 1). In every alternate iteration (whenever B gets the chance to reveal) B reveals its action (action 1) and A makes its choice of action (action 1 with highest probability) based on that. However, during A's chance to reveal, A does not reveal its action (plays action 3) and hence the agents always choose action-pair $\langle a_3, b_3 \rangle$. So, the agents choose action-pair $\langle a_1, b_1 \rangle$ whenever B's turn for revelation comes and action-pair $\langle a_3, b_3 \rangle$ whenever A's turn for revelation comes.

In the Simultaneous revelation scheme, B (having learnt to reveal) always reveals its action (action 1) and hence, A takes its best action (action 1 with highest probability) given that B has taken action 1. A has not learnt to reveal and hence never seeks to do so. Thus, both agents take action 1 and reach the desirable solution.

The question of mutual trust can be highlighted in the matrix in figure 7. If a combination of $\langle a_1, b_1 \rangle$ is being played, agent B has the incentive to change its action from b_1 to b_3 to increase its payoff from 10 to 11. When it makes such a change, A's optimal response would be to change from a_1 to a_3 to increase its payoff from 4 to 5. Thus, in their haste to respond optimally to the current situation, both agents converge to an equilibrium which pays them half of what they could have got if they had showed restraint. Each of our EUPs (in the simultaneous revelation scheme), on the other hand, trusts the other's probability-distribution over the actions (given that one of them reveals information about its action selection) and selects its action stochastically based on that distribution. Thus they progressively tend towards

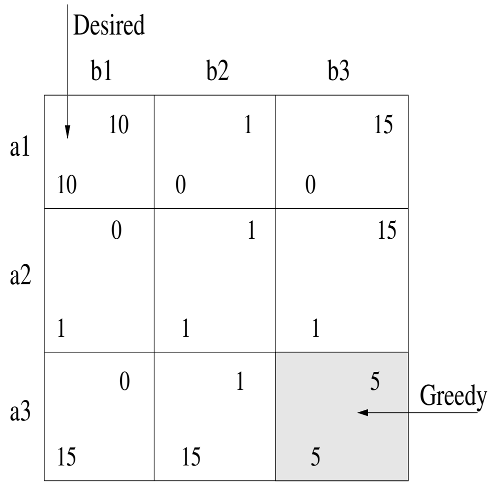

Figure 1: Game matrix where a_3 and b_3 are individually preferable to the agents, also only $\langle a_3, b_3 \rangle$ is the Nash Equilibrium.

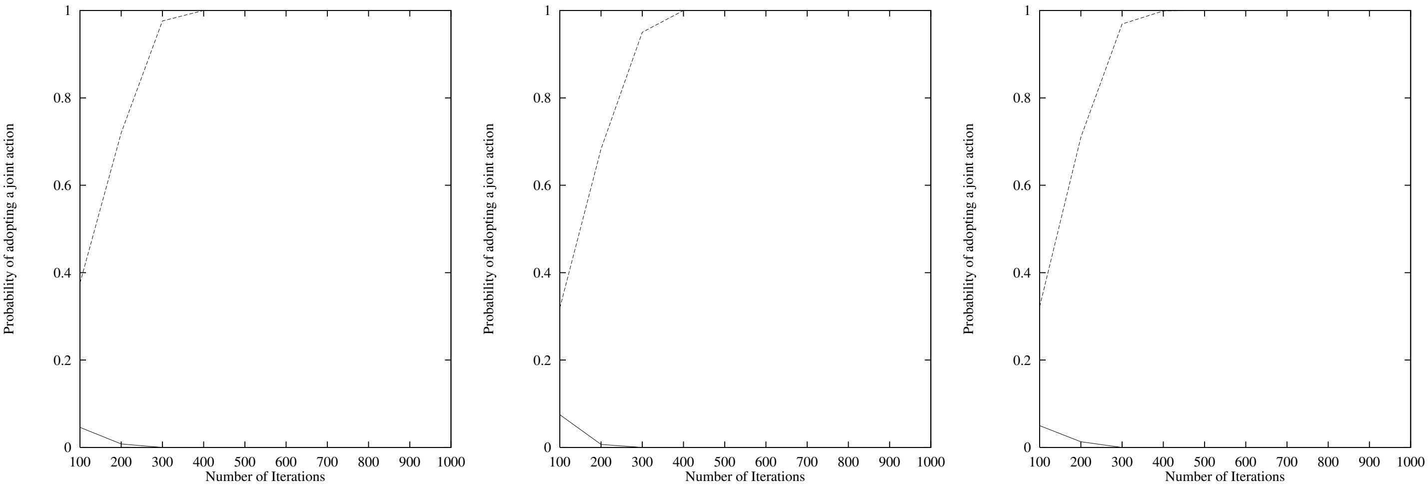

Figure 2: The probability plots for the joint actions $\langle a_1, b_1 \rangle$ (solid) and $\langle a_3, b_3 \rangle$ are shown when A and B are level-1 EUPs with no revelation, alternate revelation and simultaneous revelation (figures from left to right).

the mutually beneficial part of their search space, emulating restraint which leads to mutual benefit.

Our experimental results suggest that, in an information revealing scenario, an agent will learn to overcome its greedy (myopic) choice given the following condition - Let us consider two agents A and B. Each agent has n actions to choose from, where a_is are the actions of agent A and b_is those of agent B. Now, let a_x give the maximum expected payoff to agent A. Under this condition, agent A will have a predilection to choose action a_x during the initial exploration phase. Let us consider an iteration where agent A reveals its action to agent B. Let a_x be the chosen action for agent A. Now, agent B will choose its best response to action a_x (it will select the action which gives it the maximum average payoff given A's action). Let this action be b_y. Let R_a be the payoff to agent A due to action-pair selection (a_x,b_y). If R_a is greater than the average payoff due to the other actions that agent A can take ($R_a > \max_{w \in OA} R_w$ where OA represents other actions of agent A), the agents will learn to converge to the desirable action-pair (a_x,b_y).

5. CONCLUSIONS AND FUTURE WORK

Our basic result is that there are certain game-structures, where stochastic modeling agents can converge to high payoff points which will be missed by sophisticated modeling learners that are designed to produce Nash Equilibrium [3]. We do not tout our empirical results as an argument for always using EUPs.

Our observation, however, clearly demonstrates that learning to select a Nash Equilibrium is not necessarily the best an agent can do, and that agents who are not bound by such criteria can sometimes do better. In future, we plan to study the theoretical basis for selection of a non-equilibrium solution and identify the nature and extent of mutual trust necessary to do so.

An interesting observation from our results is that action revelation can lead to a more trusted behavior resulting in higher payoffs to the agent. In the experiment with matrix 3, agents (with action revelation) choose the more desirable Nash Equilibrium in a matrix where there are two Nash

Desired/Greedy

	b1	b2	b3
a1	10 10	9 15	0 4
a2	15 0	0 0	0 1
a3	4 0	1 0	5 5

Figure 3: Game matrix where a_1 and b_1 are relatively preferable to the agents while both $\langle a_3, b_3\rangle$ and $\langle a_1, b_1\rangle$ are the Nash Equilibria (left).

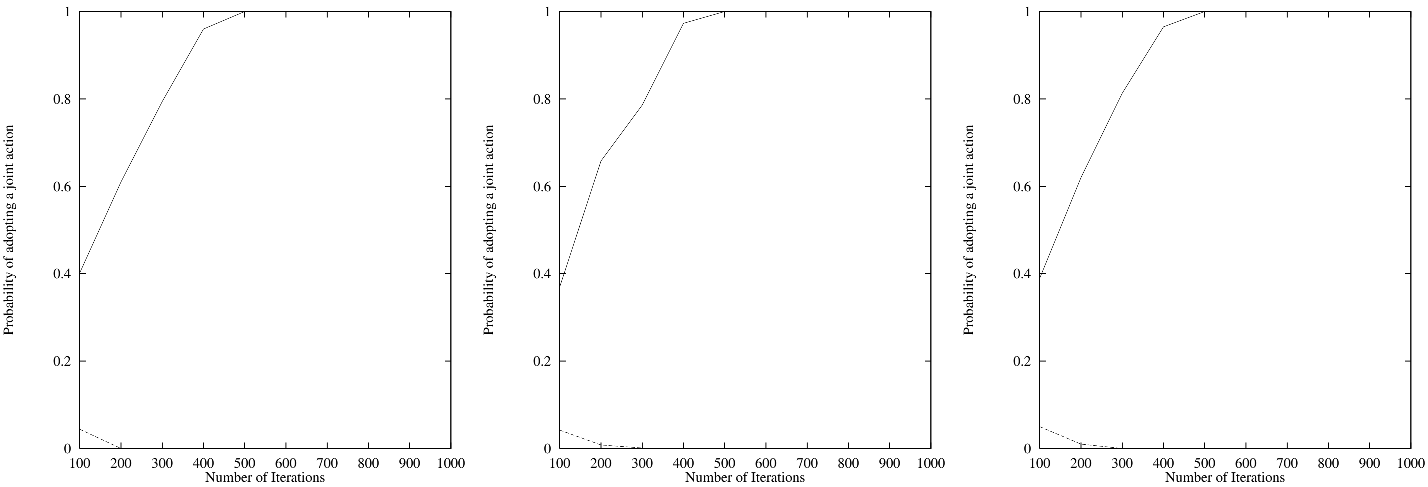

Figure 4: The probability plots for the joint actions $\langle a_1, b_1\rangle$ (solid) and $\langle a_3, b_3\rangle$ are shown when A and B are level-1 EUPs with no revelation, alternate revelation and simultaneous revelation (figures from left to right).

Equilibria. In the experiment with matrix 4, a more desirable Pareto-optimal solution is achieved as opposed to a less desirable Nash Equilibrium when Simultaneous action revelation is used. Thus, though counter-intuitive, it appears that "showing one's hand" may, sometimes, be the desirable strategy. The results also suggest that, an agent can learn to avoid revealing when the other agent tries to take advantage as shown in the experiment with matrix 4. Revealing can obviously lead to worst result for the revealer in a number of scenarios, e.g., the Prisoner's Dilemma [5]. However, we found out that both the agents learn to conceal their actions in a version of the Prisoner's Dilemma game. Our focus is to develop a strategy that allows an agent to choose its action non-myopically when the other agent reveals its action. We hope to show that such a strategy will enable agents to endure the "lure" of short term profits and may enable us to solve the iterated two-player Prisoner's Dilemma game.

Acknowledgement

This work has been supported, in part, by an NSF CAREER Award: IIS-9702672.

6. REFERENCES

[1] Steven J. Brams. *Theory of Moves*. Cambridge University Press, Cambridge: UK, 1994.

[2] Caroline Claus and Craig Boutilier. The dynamics of reinforcement learning in cooperative multiagent systems. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 746–752, Menlo Park, CA, 1998. AAAI Press/MIT Press.

[3] Junling Hu and Michael P. Wellman. Multiagent reinforcement learning: Theoretical framework and an algorithm. In Jude Shavlik, editor, *Proceedings of the Fifteenth International Conference on Machine Learning (ML'98)*, pages 242–250, San Francisco, CA, 1998. Morgan Kaufmann.

[4] Michael L. Littman. Markov games as a framework for

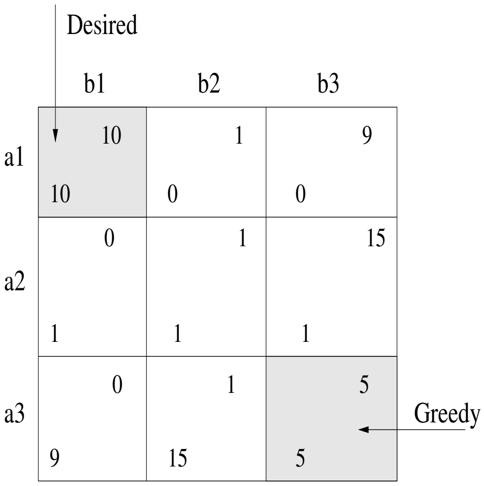

Figure 5: Game matrix where a_3 and b_3 are relatively preferable to the agents while both $\langle a_1, b_1\rangle$ and $\langle a_3, b_3\rangle$ are the Nash Equilibria (left).

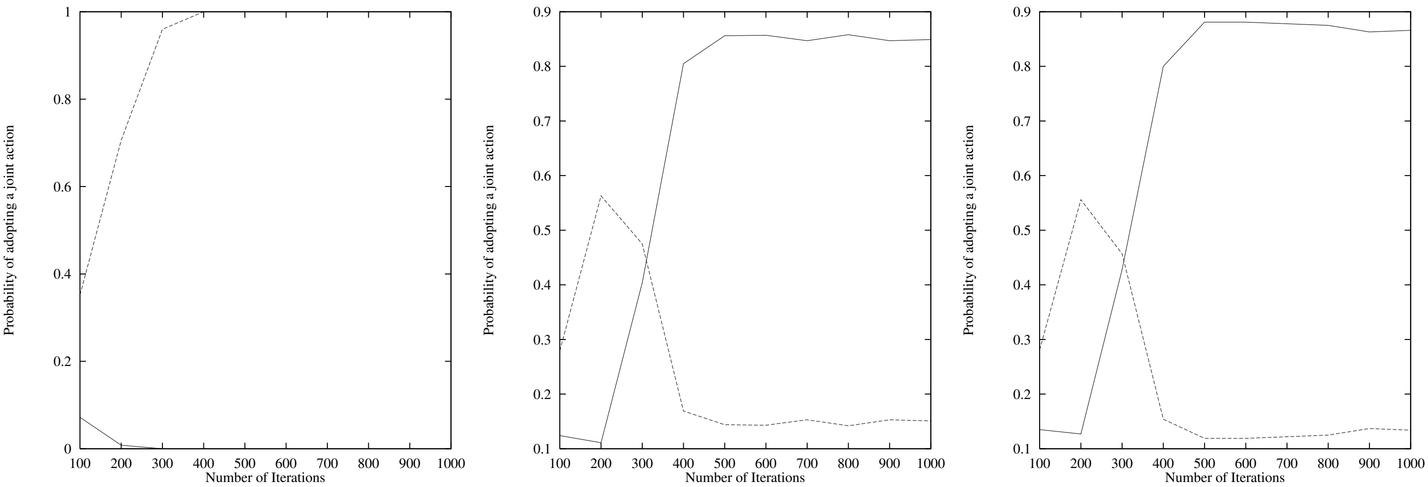

Figure 6: The probability plots for the joint actions $\langle a_1, b_1\rangle$ (solid) and $\langle a_3, b_3\rangle$ are shown when A and B are level-1 EUPs with no revelation, alternate revelation and simultaneous revelation (figures from left to right).

multi-agent reinforcement learning. In *Proceedings of the Eleventh International Conference on Machine Learning*, pages 157–163, San Mateo, CA, 1994. Morgan Kaufmann.

[5] R. Duncan Luce and Howard Raiffa. *Games and Decisions: Introduction and Critical Survey*. Dover, New York, NY, 1957.

[6] O. L. Mangasarian and H. Stone. Two-person nonzero-sum games and quadratic programming. *Journal of Mathematical Analysis and Applications*, 9:348 – 355, 1964.

[7] Manisha Mundhe and Sandip Sen. Evaluating concurrent reinforcement learners. Proceedings of the International Conference on Multiagent Systems (to appear as a poster paper), 2000.

[8] John F. Nash. Non-cooperative games. *Annals of Mathematics*, 54:286 – 295, 1951.

[9] T.W. Sandholm and R.H.Crites. Multiagent reinforcement learning in the iterated prisoner's dilemma. *Biosystems*, 37:147–166, 1995.

[10] Sandip Sen, Mahendra Sekaran, and John Hale. Learning to coordinate without sharing information. In *National Conference on Artificial Intelligence*, pages 426–431, Menlo Park, CA, 1994. AAAI Press/MIT Press. (Also published in *READINGS in AGENTS*, Michael N. Huhns and Munindar Singh (Editors), pages 509–514, Morgan Kaufmann Publishers Inc., San Francisco, CA, 1998.).

[11] C. J. C. H. Watkins and P. D. Dayan. Q-learning. *Machine Learning*, 3:279 – 292, 1992.

[12] Gerhard Weiß. Learning to coordinate actions in multi-agent systems. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 311–316, August 1993.

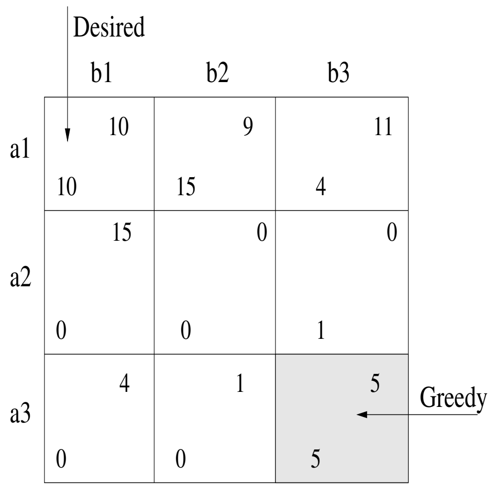

Figure 7: Game matrix where a_1 and b_1 are relatively preferable to the agents but only $\langle a_3, b_3 \rangle$ is the Nash Equilibrium (left).

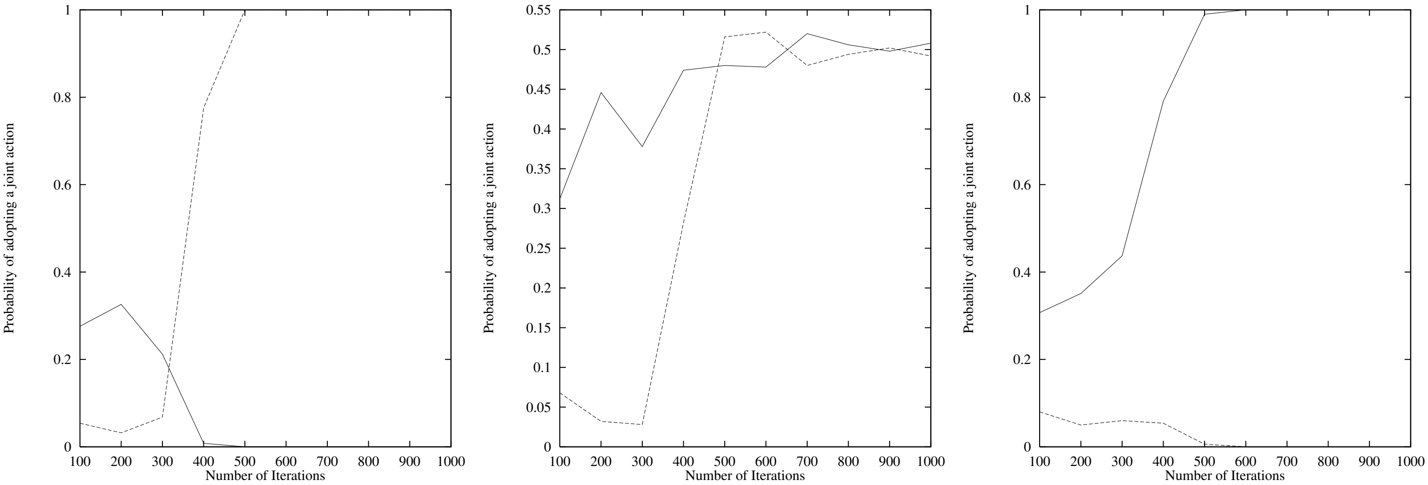

Figure 8: The probability plots for the joint actions $\langle a_1, b_1 \rangle$ (solid) and $\langle a_3, b_3 \rangle$ are shown when A and B are level-1 EUPs with no revelation, alternate revelation and simultaneous revelation (figures from left to right).