Towards a Pareto-optimal Solution in General-Sum Games

Sandip Sen
Department of Mathematics & Computer Science
The University of Tulsa
sandip-sen@utulsa.edu

Stephane Airiau
Department of Mathematics & Computer Science
The University of Tulsa
stephane@utulsa.edu

Rajatish Mukherjee
Department of Mathematics & Computer Science
The University of Tulsa
rajatish@ens.utulsa.edu

ABSTRACT

Multiagent learning literature has investigated 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 work with agents who select actions based on expected utility calculations that incorporates the observed frequencies of the actions of the opponent(s). We augment this stochastically-greedy agents with an interesting action revelation strategy that involves strategic revealing of one's action to avoid worst-case, pessimistic moves. We argue that in certain situations, such apparently risky revealing can indeed produce better payoff than a non-revealing approach. In particular, it is possible to obtain Pareto-optimal solutions that dominate Nash Equilibrium. We present results over a large number of randomly generated payoff matrices of varying sizes and compare the payoffs of strategically revealing learners to payoffs at Nash equilibrium.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence; I.2.8 [**Artificial Intelligence**]: Learning—*reinforcement learning*

General Terms

Algorithms, Experimentation, Performance

Keywords

Agents, game playing, strategy revelation

AAMAS'03, July 14–18, 2003, Melbourne, Australia.

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, e.g., 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, i.e., 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 [3, 4, 5]. This approach enables players to learn the payoffs structure of the game, provided the players can observe the opponent's action. They also ensure convergence to certain equilibria under specific assumptions. However, being able to observe the opponent's action and the structure of the game is not sufficient to reach a satisfactory outcome. In [3], 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 [3] 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. Others [4] 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. Littman [6] offers a Friend-or-Foe Q-learning approach which requires less restrictions on the play for convergence, but assumes more knowledge about the nature of the opponent, and in particular whether the opponent is cooperative or competitive in nature.

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 to future iterations of the game while selecting actions [2]. 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. In this paper we show that learning agents can 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 reach preferable outcomes by revealing their actions before the other player makes its move. 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 assume that agents are truthful about their action revelation. 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.

We present some interesting results in section 4 which 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. We also present results from a set of randomly generate matrices, that shows the relative success of revealing learners in generating superior payoffs to Nash equilibria.

2. GAMES AND EQUILIBRIA

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

DEFINITION 1. *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 Markov Decision Process (MDP) for two agents (it can be extended to n agents using n-dimensional tables instead of matrices) can be modeled 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 2. *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 3. *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*

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

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

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

Every finite bimatrix game has at least one mixed-strategy Nash Equilibria profile [11]. 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 [9]. In [4] the *ith* agent learns π_{-i}, i.e., the strategies of the other player, 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 can be seen in the Prisoner's Dilemma problem [8].

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 opponent. 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 concept adopted in the Theory of Moves approach [2]). 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. ESTIMATING PAYOFFS

A general, single-agent reinforcement learning task is an MDP, where the state transition and reward functions are unknown. A simple, model-free and on-line technique for reinforcement learning is Q-learning [13].

Our 1-level Q-learners learn Q-values[1], $Q(a, b)$, 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

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

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.

[1]For the current, stateless version of the games, the degenerate form of Q-learning with no lookahead is used. A better characterization is to view our learners as action estimators.

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 if it believes its action estimates are not correct and further exploration is needed to update them to the actual values. In this paper, however, agents use best response actions when the other player reveals. In the revealing version of the game, the players keep separate counts of the frequency based estimates of its opponent's moves for both of the following cases:

- an estimate p_b that the opponent is going to play its move b if the modeling player is not revealing its action choice, and
- an estimate $p_{b|a}$ that the opponent is going to play its move b if the modeling player reveals its choice of action a.

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, each agent maintains two different Q-tables: one corresponding to estimates of the payoff for actions when the agent reveals it, Q_r, and the other corresponding to estimates of the payoff for the same actions when the agent does not reveal it, Q_{nr}. In the following discussion, a_r refers to a revealed action and a_{nr} refers to a non-revealed action. Formally, in the exploration scheme, any action a_{nr} belonging to the set of non-revealed actions is selected with probability

$$\frac{e^{E_{nr}(a_{nr})/T}}{\sum_{a_{nr}} e^{E_{nr}(a_{nr})/T} + \sum_{a_r} e^{E_r(a_r)/T}},$$

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

$$\frac{e^{E_r(a_r)/T}}{\sum_{a_{nr}} e^{E_{nr}(a_{nr})/T} + \sum_{a_r} e^{E_r(a_r)/T}}.$$

Note that a_r and a_{nr} can take any value between 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.

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). In this case and when using *Alternate revelation choice*, in approximately 50% of the time, B will be given the opportunity to reveal its action and will not use the opportunity. With the *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

4. EXPERIMENTS

Our experimental work is presented in two stages. In the first stage, we use four game matrices to illustrate representative behaviors of EUP and revelation strategies. While these results and corresponding discussion throw some light on the characteristics of these learning approaches, it is not possible to immediately draw general conclusions about the success of these strategies when tested on arbitrary game matrices. Hence in the next experimental stage we randomly generate a large number of game matrices of different sizes and evaluate the performance of revealing strategies when compared to both pure and mixed strategy Nash equilibria payoffs in that game.

4.1 Four representative matrices

In the first stage we use 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 greedy action-profile that the agents prefer and the desirable non-greedy 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.

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 is also the Pareto-optimal solution, irrespective of the strategy chosen.
- **Matrix 3** (see figure 5) is used to demonstrate how both action revealing agents learn to choose the pareto-optimal Nash Equilibrium whereas EUPs fail to converge to this desired solution.

- **Matrix 4** (see figure 7) is used to demonstrate how Simultaneous revelation choice outperforms Alternate revelation choice, which, in turn, outperforms EUPs.

4.1.1 *Experiments with Matrix 1*

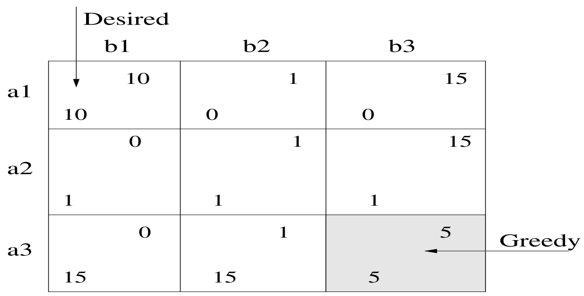

Figure 1: Game matrix where a_3 and b_3 are individually preferable to the agents, also only $\langle a_3, b_3\rangle$ is the Nash Equilibrium.

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 pareto-optimal solution, however, is for the action-combination $\langle a_1, b_1\rangle$ giving a payoff of 10 to both agents. Two EUP learners played the game for 1000 iterations using initial temperature of 10 and a temperature decay factor of 0.99. The probabilities of adopting joint-actions $\langle a_1, b_1\rangle$ and $\langle a_3, b_3\rangle$ were measured every 100 interactions as the frequencies of choosing different actions over the last 100 interactions. The values 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 a_3 and b_3 are the agents' dominant strategies. 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).

4.1.2 *Experiments with Matrix 2*

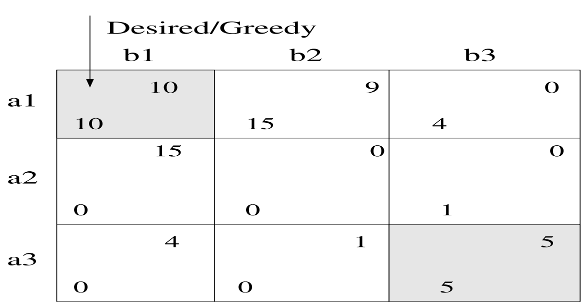

Figure 3: Game matrix where a_1 and b_1 are relatively preferable to the agents while both $\langle a_3, b_3\rangle$ and $\langle a_1, b_1\rangle$ are the Nash Equilibria (left).

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).

4.1.3 *Experiments with Matrix 3*

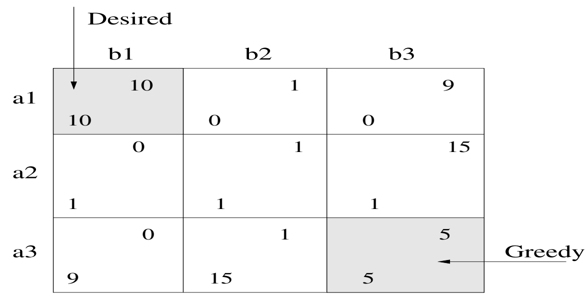

Figure 5: Game matrix where a_3 and b_3 are relatively preferable to the agents while both $\langle a_1, b_1\rangle$ and $\langle a_3, b_3\rangle$ are the Nash Equilibria (left).

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 the average payoffs for actions a_3 and b_3 are higher than actions a_1 and b_1 respectively. 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 [4] 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.

4.1.4 *Experiments with Matrix 4*

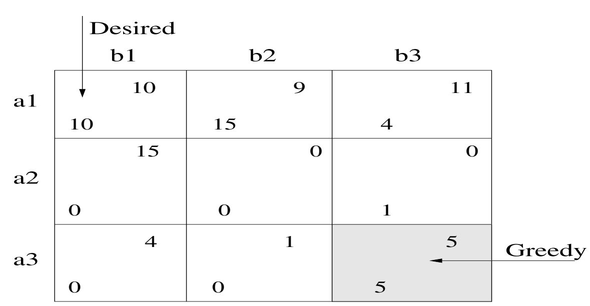

Figure 7: Game matrix where a_1 and b_1 are relatively preferable to the agents but only $\langle a_3, b_3\rangle$ is the Nash Equilibrium (left).

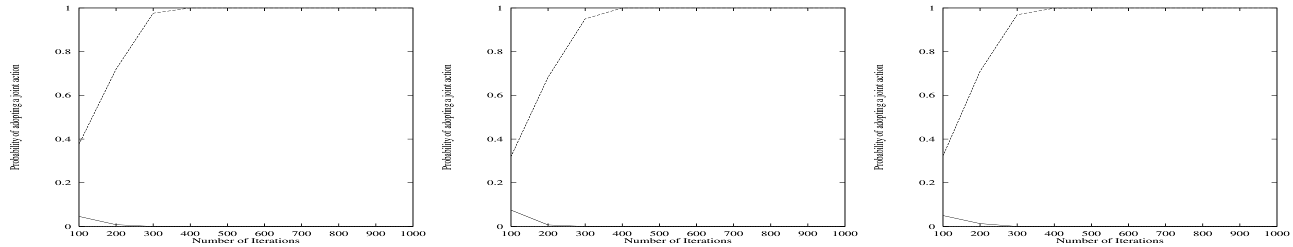

Figure 2: Probabilities for choosing joint actions $\langle a_1, b_1\rangle$ (solid) and $\langle a_3, b_3\rangle$ for Matrix 1 when A and B are level-1 EUPs with no revelation, alternate revelation and simultaneous revelation (figures from left to right).

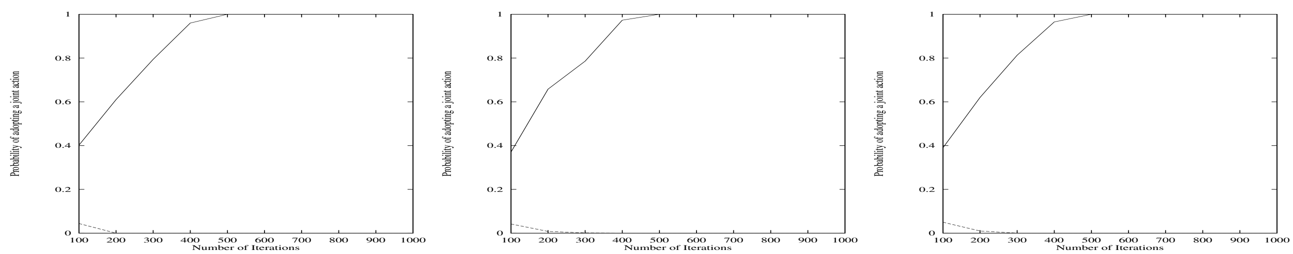

Figure 4: Probabilities for choosing joint actions $\langle a_1, b_1\rangle$ (solid) and $\langle a_3, b_3\rangle$ for Matrix 2 when A and B are level-1 EUPs with no revelation, alternate revelation and simultaneous revelation (figures from left to right).

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.

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. 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 above experiments using revelation schemes, A learns not to reveal its action: whenever A reveals action 1, B exploits A by taking action 3. B, however, learns to reveal its action 1. When using alternate revelation, in every alternate iteration, i.e., whenever B gets the chance to reveal, B reveals action 1 and A responds by choosing action 1 with high probability. 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 action 1 and A responds with its best action, i.e., action 1, with high probability. A has not learnt to reveal and hence never seeks to do so. Thus, both agents choose 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 the revealing EUPs, in the simultaneous revelation scheme, on the other hand, trusts the other's probability-distribution over the actions and selects its action stochastically based on that distribution. Thus they progressively tend towards the mutually beneficial solution in the action space, emulating restraint which leads to mutual benefit.

4.1.5 Some analysis

Based our experience with the above experiments, we outline the requirements for information revealing to generate desirable, pareto-optimal solutions. 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 want 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, i.e., 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). This is an initial, but incomplete, characterization of conditions necessary for convergence by revealing learners to pareto-optimal solutions.

4.2 Summary results over randomly generated matrices

To evaluate the general applicability of these strategic revealing based learning approaches, we generated 1000 random matrices each for agent action space sizes of 3, 5, and 7. Our goal was to find out how the payoff from solutions

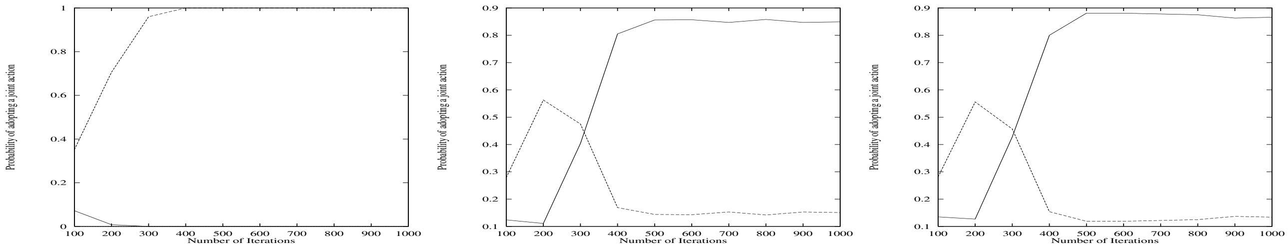

Figure 6: Probabilities for choosing joint actions $\langle a_1, b_1 \rangle$ (solid) and $\langle a_3, b_3 \rangle$ for Matrix 3 when A and B are level-1 EUPs with no revelation, alternate revelation and simultaneous revelation (figures from left to right).

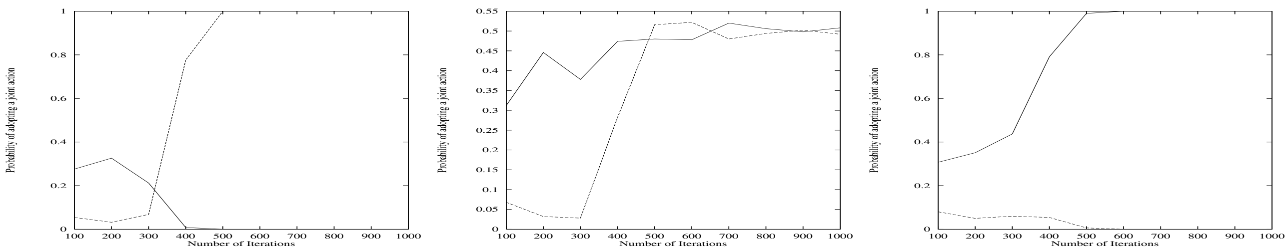

Figure 8: Probabilities for choosing joint actions $\langle a_1, b_1 \rangle$ (solid) and $\langle a_3, b_3 \rangle$ for matrix 4 when A and B are level-1 EUPs with no revelation, alternate revelation and simultaneous revelation (figures from left to right).

generated by the revealing strategies compared to Nash equilibrium payoffs.

To find out the different Nash equilibrium for a given payoff matrix, we used the Gambit game theory software (`http://www.hss.caltech.edu/gambit/`). In the following we describe our comparison metrics. We use the following notation: $NE_{i,j}$ is the set and $n_{ij} = |NE_{i,j}|$ is the corresponding number of pure and mixed strategy Nash Equilibria present in the jth of the 1000 $i \times i$ payoff matrices. $NE^k_{i,j} \in NE_{i,j}$ is the k of the n_{ij} Nash equilibria, and $NE^k_{i,j}(A)$ and $NE^k_{i,j}(B)$ correspond to the payoffs received by players A and B if they play the corresponding strategies. From the experiments presented in the last section it is clear that different runs of the revealing strategy can converge to different solutions. For fair comparison, we calculate the average of the payoffs returned by the revealing strategy over ten runs with different random number sequence on each matrix. These average payoffs for the jth of the $i \times i$ payoff matrix is referred to as $R_{i,j}(A)$ and $R_{i,j}(B)$ for players A and B respectively.

For comparison of the solutions generated by the revealing strategies with Nash Equilibrium solutions for the same matrices we report the following measurements. The basic measure is that of dominance. For a given problem size i, we partition the $i \times i$ matrices into three sets: $\{D^i_{NE}, D^i_R, D^i_=\}$ based on the following criteria

- the jth payoff matrix is a member of D^i_{NE} if the payoffs from any of the Nash equilibria solution for that matrix dominates the average payoff from revealing solutions, i.e., $\exists x \in NE_{i,j}$ s.t. either $x(A) \geq R_{i,j}(A)$ & $x(B) > R_{i,j}(B)$ or $x(A) > R_{i,j}(A)$ & $x(B) \geq R_{i,j}(B)$.
- the jth payoff matrix is a member of D^i_R if the payoffs from all of the Nash equilibria solutions for that matrix are dominated by the average payoff from revealing solutions, i.e., $\forall x \in NE_{i,j}$ either $x(A) \leq R_{i,j}(A)$ & $x(B) < R_{i,j}(B)$ or $x(A) < R_{i,j}(A)$ & $x(B) \leq R_{i,j}(B)$.
- the jth payoff matrix is a member of $D^i_=$ if none of the above conditions hold.

Let $p^i_{NE}, p^i_r, p^i_=$ be the corresponding percentage of matrices where some Nash equilibria dominate the average revealing based payoff, the average revealing based payoff dominates all Nash equilibria payoffs, and when the Nash equilibria and the revealing based payoffs do not dominate each other. We plot these percentages, over 1000 randomly generated matrices each, for problems of size 3, 5, and 7 in Figure 9. We find that for the smallest problem size, i.e., 3 actions per agent, the revealing strategy do dominate the Nash equilibrium payoffs more often. As the problem sizes increase though, the revealing strategy gets dominated more often. The number, but not the percentage, of "no dominance" situation also decreases. It was still surprising for us to observe that such a simple revealing strategy can end up dominating Nash equilibrium payoffs in a significant number of scenarios.

Note that the above comparison is somewhat biased against the revealing strategy. Whereas we take the average payoff from the different runs with the revealing strategy, we do not consider the average of the Nash equilibria payoffs. Learners that converge to Nash equilibria have no guarantee of converging to a dominant Nash equilibria. Another comparison metric can be to see how many of the individual Nash equilibrium solutions dominate the average revealing payoffs and vice versa[2]. We also report the cases when neither dominate. The results are plotted in Figure 10 for different

[2]We note that the total number of Nash Equilibrium in the 1000 randomly generated matrices grows with the size of the matrices. The corresponding numbers for problem sizes 2, 5, and 7 are 1871, 2682, and 4692. Therefore the average number of Nash equilibria per matrix grows from approximately 1.87 to 2.68 to 4.69 as the action space of agent grow from 3, 5, 7. These numbers depend, of course, on the assumption of uniformly generated random numbers to designate payoffs.

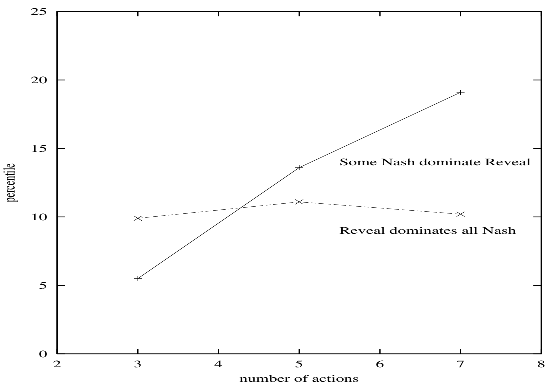

Figure 9: Percentage of matrices where some Nash equilibrium solution dominated the average revealing payoff and the average revealing payoff dominated all Nash equilibrium solutions on the same matrix.

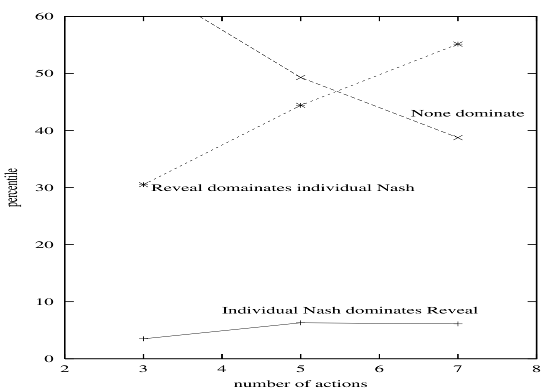

Figure 10: Percentage of Nash equilibrium solutions that dominated, was dominated, or was incomparable to the average revealing payoffs on the same matrix.

problem sizes. We now see that the average revealing payoff dominates a significantly larger number of Nash equilibrium payoffs and the frequency of domination grows with larger problem sizes. The number of non-dominating cases grows at a lesser rate, and hence the corresponding percentage decreases.

We next analyzed the non-dominated payoff cases. We wanted to find out whether revealing strategies produced greater social benefits in these cases, i.e., whether the sum of the payoffs to the two agents is higher with the revealing approach compared to Nash equilibrium solutions. We plot these sums in Figure 11. We found that for non-dominated cases, the average revealing payoff sum is almost twice as likely to be greater than the sum of Nash equilibrium payoff on the same matrix. Figures 11 and 10 combined supports the argument that revealing learners are likely to produce larger and more dominant social welfare compared to learners that converge to some Nash equilibrium solutions.

We now revisit our argument that it is somewhat unfair to compare the average revealing payoff with the best Nash equilibrium payoff. Rather, for a given matrix j for

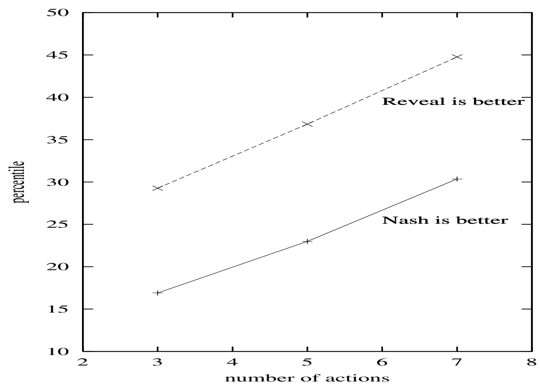

Figure 11: Percentage of Nash equilibrium solutions where the sum of payoffs to both agents from it was greater or less than the sum of the average payoffs obtained by revealing strategy on the same matrix when neither the Nash equilibrium payoff nor the average revealing payoff was dominant.

a problem size i we can compare $\frac{1}{n_{ij}} \sum_{k=1}^{k=n_{ij}} (NE^k_{i,j}(A) + NE^k_{i,j}(B))$, i.e., the average of the two payoffs over all Nash equilibrium for the matrix, with $R_{i,j}(A) + R_{i,j}(B)$, the sum of the average payoff from the revealing strategy for the same matrix. We present in Figure 12 the percentage of times the average of the sum of the payoffs from all Nash equilibria in a matrix is more or less than the sum of the average payoffs obtained by the revealing approach. We see that in this, somewhat fairer metric, the revealing approach significantly outperforms the Nash Equilibrium solutions, and this performance difference grows for larger problem sizes. The percentages do not add up to 100, because in some cases the payoffs produced by both approaches are the same. Note though that in this case, we are evaluating dominance from the social welfare point of view, and not from the individual selfish payoff perspective. One can argue, however, that this observation may increase the viability of the revealing strategy if the possibility of side payments is accepted, i.e., agents can negotiate to subsequently reallocate larger payoffs obtained by playing the revealing strategy.

5. DISCUSSIONS

Learning in the context of iterated bimatrix and stochastic games have received considerable attention in the multiagent learning literature [1, 4, 6]. The focus of these work is on convergence to rational play. An orthogonal line of research have addressed the problem of improving payoffs over Nash equilibria solutions. One of these methods use aspiration levels and does not require a learner to know about the choices or payoffs of the opponent [12]. This method, however, is critically dependent on the choice of appropriate aspiration levels. Another method uses stubbornness and threats as implicit communication to lead the opponent to desirable solutions [7]. This method, however, requires the knowledge of both opponent action and the payoffs they receive. In this work, we assume that only the actions taken by the opponent, and not the payoffs they receive, are observable.

Our basic result is that there are certain game-structures, where apparently harmful (as revealing actions opens up

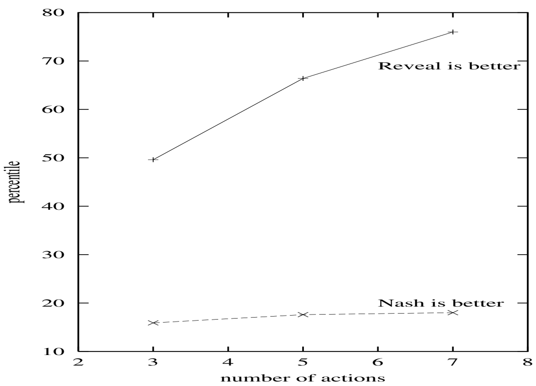

Figure 12: Percentage of matrices where average of the sum of payoffs to both agents over all Nash equilibriums of a matrix was greater or less than the sum of the average payoffs obtained by revealing strategy.

possibilities for best response play by opponent which can hurt the revealer's interests), action-revealing agents can converge to high payoff solutions which will be missed by sophisticated modeling learners that are designed to produce Nash Equilibrium [4]. Our results from both sample matrices and average results over a large set of randomly generated matrices demonstrate that a simple revealing based learning strategy can consistently provide better rewards than what can be obtained with learners trying to achieve Nash equilibrium. Together with the lack of any method guaranteed to learn Nash equilibrium [10] in general-sum games, the current results can be used as a motivation to study alternative goals for learning agents. In future, we plan to study the theoretical basis for selection of a non-Nash equilibrium solution and identify the nature and extent of mutual trust necessary to do so.

We plan to study the converge behavior of the revealing learner with the goal of characterizing when such learning behavior will produce Pareto-optimal solutions. A secondary goal in this process of study will be to identify possible improvements to the current revealing mechanism.

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 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 [8]. 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. The strategy of best response to revealed action used in this paper is a greedy one. We plan to work on developing non-myopic strategies with strategic look-ahead in iterated games and show that such a strategy will enable agents to endure the "lure" of short term profits and may enable us to solve difficult problems like the iterated two-player Prisoner's Dilemma game.

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

6. REFERENCES

[1] Michael Bowling and Manuela Veloso. Rational and convergent learning in stochastic games. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 1021–1026, 2001.

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

[3] 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.

[4] 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*, pages 242–250, San Francisco, CA, 1998. Morgan Kaufmann.

[5] Michael L. Littman. Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the Eleventh International Conference on Machine Learning*, pages 157–163, San Mateo, CA, 1994. Morgan Kaufmann.

[6] Michael L. Littman. Friend-or-foe q-learning in general-sum games. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 322–328, San Francisco: CA, 2001. Morgan Kaufmann.

[7] Michael L. Littman and Peter Stone. Implicit negotiation in repeated games. In *Intelligent Agents VIII: AGENT THEORIES, ARCHITECTURE, AND LANGUAGES*, pages 393–404, 2001.

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

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

[10] J. Nachbar. Prediction, optimization and learning in repeated games. *Econometrica*, 65:275 – 309, 1997.

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

[12] Jeff L. Stimpson, Michael A. Goodrich, and Lawrence C. Walters. Satisficing and learning cooperation in the prisoner's dilemma. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 535–540, 2001.

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