Q Learning and Fees Relating to The Santa Fe Bar Problem

Julie Farago and Amy Greenwald

Please do not cite or distribute without permission of authors

1 Abstarct

In this paper we tackle the questions of learning verse rationality in the the Santa Fe Bar Problem. Using a combination of learning techniques we attempted to coherce our agents into learning both a fair and efficient solution to the problem. Though we did not find a definitive solution to the Santa Fe Bar Problem, we did make substantial progress by taking new approaches to an old problem and by restructuring the problem into a new and more workable situaion.

2 Introduction

The *Santa Fe Bar Problem* (SFBP) was introduced by Brian Arthur, an economist at the Santa Fe Institute. Using this model, Arthur challenges the behavioral predictions of rational choice theory, since, in the SFBP, *rationality precludes learning*. Bounded rationality, however, is not inconsistent with adaptive learning. Here is the scenario:

N [(say, 100)] people decide independently each week whether to go to a bar that offers entertainment on a certain night . . . Space is limited, and the evening is enjoyable if things are not too crowded – especially, if fewer that 60 [or, some fixed but perhaps unknown capacity c] percent of the possible 100 are present . . . a person or agent goes if she expects fewer than 60 to show up and stays home if she expects more than 60 to go, Choices are unaffected by previous visits; there is no collusion or prior communication among the agents; the only information available is the number who came in the past weeks.

The bar in Santa Fe is called a *congested* resource, because the value to an agent for attending the bar depends on the number of other agents that attend the bar. In other words, the agents' valuations of congested resources are not idepentent to the agent, rather, they are exogenously-determined.[1] This is similiar to many real-world situations, ranging from fishermen fishing in common waters, to farmers polluting common water supplies, to network users monopolizing bandwidth, to other versions of the tragedy of the commons that are characterized by negative externalities.[2]

Following Arthur, let us analyze SFBP under the standard assumption of *rationality: i.e.* utility-maximization with respect to one's beliefs. Define an *uncrowded* bar as one in which attendance is less than or equal to the capacity, c, and define a crowded bar as one in which attendance is strictly greater than c. Let the utility of going to an uncrowded bar be 1 and the utility of going to a crowded bar be -1; the utility of staying home is 0, regardless of the state of the bar. In this setup, rational choice theory dictates the following behavior: if an agent believes that the bar will be uncrowded with a probability p, then his rational choices are to go to the bar if $p > 1/2$ and to stay home if $p < 1/2$. (In the case where p=1/2, rational agents are indifferent between attending the bar and staying home and may have arbitrarily: *i.e.* randomize.) The probability p that the bar is uncrowded is determined by the agents (possibly randomized) strategies. If all the agents learn the correct value of p, then we stumble upon the fundamental contradiction. Suppose the agents learn that the bar will be uncrowded with probability $p < 1/2$; then in fact the bar will be empty with probability 1. On the other hand, suppose the agents learn that the bar will be uncrowded with probability $p > 1/2$; then the bar will be empty with probability 0.[3,4] The only conclusion is that rational agents cannot learn in SFBP; therefore, *rationality precludes learning*.

The Santa Fe Bar Problem is a repeated game. In Arthur's original paper, he demonstrated via simulations that certain types of *boundedly rational* agents are capable of learning to center their collective attendance around the capacity of the bar. Arthur's approach is based on complex modeling of cognitive aspects of inductive reasoning. In Greenwald [3], boundedly rational, no-regret learning agents are simulated and shown to converge to the neighborhood of the symmetric mixed strategy Nash equilibrium in which all agents attend the bar with probability $\sim c/N$. In our current research we tried a number of different learning algorithms, including hedge-boost and Q-learning, converging to probabilistic and deterministic strategies.

[1] determined by an outside source

[2] An externality is a third-party effect. An example of a negative externality is pollution. An example of a positive externality is standardization

[3] Schelling [8] refers to phenomena of this kind as self-negating prophecies.

[4] Yogi Berra also observed this phenomenon. He said, "Nobody goes there anymore; it's too crowded."

3 Goals

Our goals for the Santa Fe Bar Problem are two-fold. The game should be *fair* and *efficient*. In a *fair* game, all of the agents attend the bar an equal number of times over an extended time period. In an *efficient* game, the agents maximize the total utility each round. If there are 100 agents who all want to attend the El Farol bar, which can only comfortably hold 60 people, than in a fair and maximized solution 60 agents attend the bar every night and all agents go to El Farol 60% of the time.

4 Equilibrium Analysis

The mixed strategy Nash Equilibrium for the Santa Fe Bar Problem with ($c = 60$ and $N = 100$) sets each agents' probability of going to the bar, p, to .6 (c/N).

When each agent's probability of going to the bar, p, is set to .6, mean bar attendance becomes 60 with a variance of 24 (standard deviation is thus about $\pm$ 5). The agents attend the bar equally, but the average utility converges to around 0 to because of the variance above and below 60. This solution is therefore fair but inefficient, as seen in Figure 4. Additionally shown by the table of probability verse predicted utility, small variations in p around c/N result in large variations in the cumulative utility of the game.

p	Expected Utility	p	Expected Utility
0	0.0	.54	41.9
.1	10.0	.56	33.0
.2	20.0	.58	18.8
.3	30.0	.6	0.5
.4	40.0	.7	-67.5
.46	45.8	.8	-80.0
.48	47.2	.9	-90.0
.5	47.8	1.0	-100.0
.52	46.5		

Figure 1: The table on the shows the expected utility per game when the agents probability of going to the bar is set to p. When this value is set to c/N, or .6, the expected utility converges to about 0.

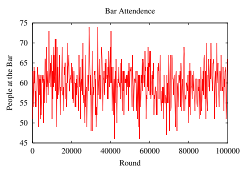

Figure 2: The graph shows the attendance at the bar when p is set to .6; this attendance oscillates around 60 with an average amplitude of about 5 (the standard deviation).

4.1 Standard Learning Problems

When attempting to create agents who learn a fair solution to this problem that results in a high positive utility, two standard problems arise. The agents either converge to an unfair solution, in which some agents go to the bar all the time while others never attend, or they achieve an inefficient solution (like the mixed Nash Equilibrium). The research done on the SFBP therefore progressed in three stages. To begin, a form of boosting, Hedge Boost [2], was implemented because the algorithm causes all the agents to strongly want to take the action that has a history of higher rewards. Due to the lack of success, Hedge Boost was abandoned for q learning. In the second stage of research, the agents trained as q learners with multiple states in order to encourage cooperation. Because of the positive results from stage two, the third research phase retained the q learning while introducing a fee for attending the bar, the profits of which would be redistributed among the agents who stayed home.

5 Stage One: Hedge, a Form of Boosting

Boosting improves upon the performance of a weak learning algorithm. A *weak* learning algorithm performs slightly better than a random classifier. For example, if an agent tried to classify a mixture of 50 baseballs and 50 softballs correctly, based on a few key attributes like size and color, and classified 60 (or 60%) correctly, then it performed slightly better (verse 50%) than a random classification. In the above case of baseballs and softballs, after adding a boosting algorithm, the agent might then classify 85% of the balls correctly.

Hedge Boost [2], the particular algorithm implemented, uses exponentiation to emphasize small differences in cumulative rewards. An agent uses the new, boosted results, here expressed as q_i, to probabilistically choose an action from her action set. Define $\Pi_i^t(s_i)$ as agent i's cumulative rewards for taking action s_i, from the set S_i, through time t. The formula that calculates q_i at time $t+1$ is:

$$q_i^{t+1}(s_i) = \frac{(1+\alpha)^{\Pi_i^t(s_i)}}{\sum_{s_i' \in S_i} (1+\alpha)^{\Pi_i^t(s_i')}} \quad (1)$$

[5]

In order to ensure that agents will not simply converge to the first action that gives them a reward, but rather explore to find the best strategy, an exploration rate of ϵ was added, where $\epsilon = .1$. Because of the exploration factor, agents using hedge act probabilistically on a new $\hat{q}$ instead of just q. Define $\hat{q}$ as:

$$\hat{q}_i^t = (1-\epsilon) q_i^t + \frac{\epsilon}{|S_i|} \quad (2)$$

[4] All of the agents trained with Hedge have the same S. Let $S = \{home, bar\}$.

5.1 Fully Informed Agents

As a base to build off and shoot for, tests were first run with fully *informed* agents. An informed agent updates her cumulative reward, Π, for every action every round. For example, if agent Anna decides to stay home the first round, and the bar happened to be uncrowded, her $\Pi_{bar}^1(s_{bar})$ would be equal to 1 (even though she never actually attended to bar her Π_{Bar} field is still incremented). Informed agents are, in some

[5] For our experiments $\alpha = .05$.

sense, omniscient agents. Therefor the cumulative reward update rule for an informed agent is:

$$\Pi_i^{t+1}(s_i) = \sum_{x=1}^{t} \rho_i(s_i, s_{-i}^x) \tag{3}$$

Where ρ is the reward for taking action s at time t.

The fully informed agents should in theory, and did in practice, converge to a mixed Nash Equilibrium. Though this is a fair solution, it struggles from two major problems. First, the average utility converges to 0 because this is the mixed Nash Equilibrium, second, the setup is unrealistic, as an omniscient agent rarely exist. Because fully informed agents are virtually non-existent outside of theoretic games, the next step sought a similar outcome with a more realistic setup.

5.2 Naive Agents

A *Naive* agent only receives rewards for the actions they take. Naive agents are far more likely to exist in the real world as compared to their informed counterparts, as a person who chooses to stay home one night would be unlikely to know whether the bar had been crowded or uncrowded that evening. For a naive again, let Π be set by equation 4

$$\Pi_i^{t+1}(s_i) = \sum_{x=1}^{t} \hat{\rho}_i(s_i, s_{-i}^x) \tag{4}$$

[4] Additionally, the reward is only given if an agent takes an action which results in a payoff. Unlike our informed agent, Anna, who received all the rewards no matter her action, a Naive agent, Lynda, would only get +1 reward for attending the bar if she actually went there (and, of course it was uncrowded). The reward function for our naive agent is therefore:[6]

$$\hat{\rho}_i(s_i, s_{-i}^x) = \begin{cases} \frac{\rho_i(s_i, s_{-i}^t)}{\hat{q}_i^t(s_i)} & \text{if} s_i^t = s_i \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Unlike the informed agents, the naive agents converge to a pure Nash Equilibrium, where some agents go to the bar all of the time and some never go. The average utility of this game converges to around 60. Though this game is efficient, it is also unfair and therefore undesired. Looking at these results we determined that the agents required a little more ingenuity to attain our ultimate goals.

6 Stage Two: Q-Learning

The Santa Fe Bar Problem is generally approached assuming a boolean set of actions, like in Hedge. Yet, often when there are only two possible actions, agents learn unfair strategies. Although the overall utility is maximized, one set of agents always goes to the bar, while the rest stay home. This is not the desired result.

The SFBP can be viewed as a problem with states as well as actions. The basic Q matrix has two states, Home and Bar, each of which can transition either back to itself or to the other state. This additional freedom allows for movement that was not possible in the previous example. Additionally, the combination of a high exploration rate and high emphasis on the future (expressed in terms of ϵ and γ) results in convergence to a mixed Nash Equilibrium, as will be shown later. [7]

Furthermore, when thinking about the best possible outcome (a fair game in which the overall utility of the game is maximized) it is natural to imagine a certain amount of coordination between agents. In order for the agents to coordinate, a third action was added, in this case the *wait* action. By allowing the agents to wait and go to the bar the next round (guaranteed attendance), the agents can begin to work together to maximize not just their own, but also the overall utility of the game.

6.1 The Basic Q Learner

The Q learning update rule used was:

$$Q(i,a) \leftarrow (1-\alpha)Q(i,a) + \alpha(\rho + \gamma max_b\, Q(j,b)) \tag{6}$$

[9]

Where i is the current state, a the previous action, j the new state, b the set of all possible actions from state j, ρ the reward, gamma is the weight put on the future, and α the learning rate. α is set according to Equation 7

$$\alpha_n = \alpha_0 \left(\frac{n_0 + 1}{n_0 + n} \right)^8 \tag{7}$$

[1]

6.2 Q-Learning With Two Possible Actions

The two state Q matrix has a Home and a Bar state. Both states have the same set of possible actions; one of these actions takes the agent to the Bar state and the other takes the agent to the Home state. The reward for the Home state is always 0. The reward for the Bar state is $\pm\, \rho$ depending on whether or not the bar is crowded.[9] See Figure 6.2.

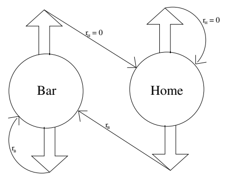

Figure 3: A graphical representation of the Q state/action transitions

[6] Note that we used $\hat{q}$ to ensure and ϵ amount of exploration

[7] Note that all agents trained by the q learning algorithm are Naive

[8] In our experiments α_0 was set to 1, n_0 was set to n (the number of rounds)for the continuous learners and n/100 for the discrete learners.

[9] Recall that in the standard problem ρ is set to 1

6.3 Q-learning With Three Possible Actions

The three state learner has an additional state, dubbed the *wait* state. The only action possible from the wait state is to go to the bar. Thus, any agent that chooses to wait is guaranteed bar attendance in the following round. Because the agents learn the Q-matrix and set rewards with a fair amount of back-propagation, the agents learn to wait because it ensures future rewards. Figure 6.3 displays the setup of the world, in action transitions.

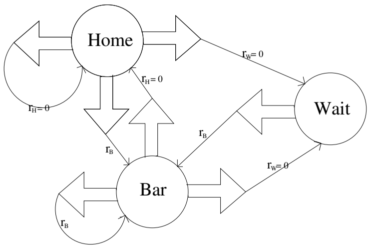

Figure 4: The Three State Q Learner setup.

The Q learner can be trained either by exploring (i.e. acting randomly) completely first, and then only exploiting on the knowledge it gained[10], or by having its exploration rate decay while the exploitation rate escalates. These two training methods lead to substantially different results.

6.4 Exploration Followed By Exploitation

The agents that explore completely before exploiting act randomly for the first 20%, in our simulations 20,000, of the rounds and then exploit for the rest of the time (80,000 rounds). The two state q learners attend the bar about 50% during the rounds of exploration. Next they experience a brief fluctuation period in which they learn their deterministic state-action matrix. Finally the agents simply follow the actions determined by the Q matrix.

Experimental results showed that each of the agents converges to one of three possible action sets: They either go to the bar everyday, switch off between the bar and home states, or always stay at the home state. This convergence occurs during the fluctuation period. Since the agents all act completely randomly for the first 20,000 rounds, the values within their Q matrices should also be reactively random, and thus not particularly good to learn on (as they are learning against other agents random actions instead of other agents deterministic actions). During the period of fluctuation, all of the agents act deterministically and so they quickly converge to a strategy with the highest rewards.

Interestingly, when the exploration period at the beginning is too small, the agents do not perform as well as they do with a longer initial exploration period; they converge to both an unfair and inefficient solution.

The two state q-learners are an improvement on the Naive Hedge, as they converge to a solution which is slightly more fair than just having agents either always go to the bar or always stay home while still retaining a high efficiency. See Figure 6.4. Unfortunately, it remains far from ideal as there are still agents who never attend the bar.

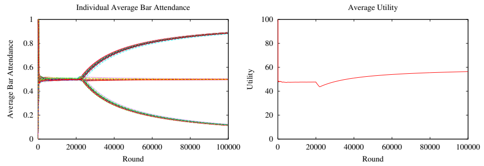

Figure 5: The graph on the left plots the average number of times each agent attends the bar over 100,000 rounds. The agents segregate themselves into three groups, which go to the bar every round (top line), every other round (middle line), or never (bottom line). The right figure shows the average utility for the game, which approaches 60 as time progresses

In an attempt to coerce the agents into a fair and efficient solution, we next trained them as three state Q-learners. The training process was the same, exploring for the first 20% of the rounds and then exploiting the knowledge gained. These agents improve upon the fairness of the 2-state game without sacrificing overall utility (visible in Figure 6.4). When there are three states all of the agents learn to go to the bar at least every other day. And, while this is not a completely fair solution, it is a big step. With this *semi-fair* solution, in the worst case, an agent goes to the bar 50% of the time, or 83% of its optimal number as compared to not going at all. Far more importantly, every time it goes, the agent receives positive utility. So, while the agent may have slightly better bar attendance with a mixed Nash equilibrium, her utility is far better in this *semi-fair* solution.

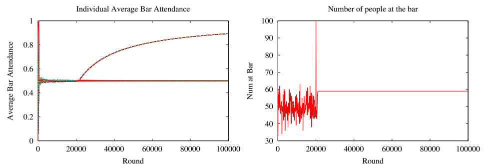

Figure 6: As with their predecessors, these three state discrete q learners maximize their utility by converging to a bar attendance of 60 (right). Yet, unlike their predecessors, all of the agents trained on three states attend the bar at least 50% of the time. This is shown in the left graph, where some agents attend the bar daily (the upper line), while the others (the lower line) attend every other day.

Unfortunately, additional states (beyond three) did not improve upon the 3 state standard. The agents, instead of converging to a fair solution, ended up branching out on all the possible state action paths. Thus, if there were four states some agents would go 1/4 of the time, which is clearly worse that 1/2. Even when the state setup was custom made to the scenario, with three bar states and two non-bar states, the agents did not converge to a 60% bar attendance.

[10]Note that the Q Learner continues to learn after it is done exploring

6.5 Exploration Decaying into Exploitation

The second set of agents trained, whose exploration decays as a function of time, succeed in generating a fair solution, but only accomplished this goal at the cost of the overall utility. The exploration rate decreases according to Equation 8.[11]

$$\epsilon_n = \epsilon_0 \left(\frac{n_0 + 1}{n_0 + n} \right)^{12} \quad (8)$$

This set of decaying exploration Q learners, or *continuous* Q learners, converges to the mixed Nash equilibrium (as seen in in Figure 6.5), and, as would be expected, the average utility approaches 0.

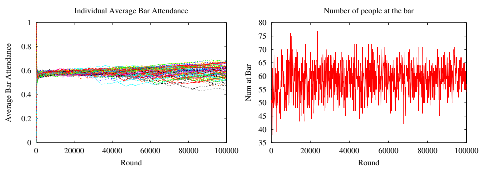

Figure 7: As shown in the left graph, a fair solution is reached by the continuous learner, as the cumulative bar attendance for all of the agents is virtually the same. The right figure displays actual bar attendance at different round, which oscillates around 60.

When analyzing the continuous Q learners and their actions, one must notice that the results being displayed are *not converging*. Because Alpha decays so slowly, the result tends to stay closer to the Nash Equilibrium. This happens because there agents place a lot of weight on their most recent results without paying much head to the past. Thus, their actions are always changing. When looking at the Q matrices and their values one can see this fluctuation more clearly. Different actions take precedence at different rounds, without converging to a set action plan. See Figure 6.5.

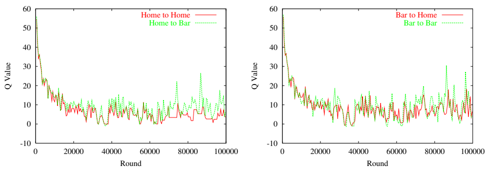

Figure 8: The Left figure shows the changing Q values in the home state for the bar and home actions. The action to stay home from the home state is very rarely chosen, as it is rarely the top line (higher value). On the other hand, the left graph shows the Q values for the bar state, which clearly switch off between the go home and go to the bar actions.

Unlike with the discrete Q learnering agents, adding extra states to the continuous learners made little difference and did not improve the average utility of the game. Because the possibility for more coordination did not bring us closer to our goal of a high efficiency game, we decided to go another route.

7 Stage Three: Add A Fee

In addition to the need for coordination between agents, another major problem beneath the SFBP learning paradox is the lack of a strong incentive to stay home. We therefore chose to fill this void. In order to add an incentive to stay home, while not changing the basic structure of the problem, we added a *fee*, χ, for attending the bar. χ varies vary over the range of $[0,\rho]$.[13] Thus, the reward for attending the bar is changed to:

$$'\rho = \begin{cases} \rho - \chi & if\ n <= c \\ -\rho - \chi & if\ n > c \end{cases} \qquad n \Rightarrow \text{number of people at the bar}$$

Those agents who remain home redistribute the sum collected from the fee evenly amongst themselves. Therefore their new reward (for staying home) is :

$$\rho = \frac{n\chi}{N - n} \quad \text{where } N \text{ is the total number of agents}$$

χ both decreases the incentive to go to the bar all of the time and increases the incentive to stay home; enhancing and encouraging convergence to an Equilibrium. The goal of the fee is to encourage a fair result while retaining good overall utility (unlike the strictly mixed Nash equilibrium).

7.1 Equilibrium Analysis

In order to see how a fee actually effected the predicted equilibrium, we had to solve equation 9 for the different possible values of χ. Equation 9 sets the total possible rewards received for going to the bar (in all the different combinations of players) equal to the total possible rewards for staying home. While the reward function for going to the bar changes depending on the capacity and load, the reward function for staying home remains constant (that is why there are two summations on the left and only one on the right). This equation reflects the reward functions explained about, and are summed over N, as there are N possible situations.

$$\begin{aligned} (\rho - \chi) \sum_{i=0}^{c} \binom{N}{i} p^i (1-p)^{N-i} + (-\rho - \chi) \\ * \sum_{i=c+1}^{N} \binom{N}{i} p^i (1-p)^{N-i} \\ = \chi \sum_{i=0}^{N} \binom{N}{i} p^i (1-p)^{N-i} \frac{i}{N-i} \end{aligned} \quad (9)$$

Since we are dealing with the standard SFBP, let N be 100 and c be 60. When we set ρ to 1 and vary χ over [0:1] (inclusive) the equilibrium decreases slowly from $\chi = 0$ to $\chi = .4$, and then rapidly declines until $\chi = 1$. For a graphical view of this see Figure 7.1

[11]Note that this is the same equation as used for the alpha decay from Equation 6

[12]In our experiments ϵ_0 was set to 1, n_0 was set to n (the number of rounds)/100 for the continuous learners.

[13]Recall that ρ is the reward for attending an uncrowded bar

When the 2 state continuous agents were trained on varying fees, their equilibrium closely follows the predicted equilibrium. Again, see Figure 7.1

Because there are many different possible cases for the general Santa Fe Bar Problem (i.e. maybe 563 agents want to go to the bar, but are unhappy when there are over 467 people there), we did not want to simply determine what the best fee was and hard code it in. On the other-hand, a far more useful fee determining method would be some sort of convergence. Because of this we choose to implement a derivative follow algorithm for the fee change and convergence.

7.2 An Iterative Mechanism

In Arthur's original formulation of SFBP, it is assumed that all agents have identical preferences, and moreover that these preferences are known. Now suppose that the value to each agent of going to the bar is unknown to a central planner, say the mayor, whose goal is to design a mechanism, the equilibria of which maximize the total utility of the townspeople. Assume each agent i's utility u_i is an endogenous function of the other agents' actions: let u_i be a concave function of the attendance at the bar, say λ, and let μ_i be the point at which agent i's utility peaks: *i.e.*, $u_i(\lambda) = \max\{1 - (\lambda - \mu_i)^2/\sigma_i^2, 0\}$ for $\sigma_i \geq 0$. In this setting, total utility is maximized at the median value of the μ_i's, say μ_i^*, with those agents whose valuations are closest to μ_i^* attending the bar. Thus, if 25 agents' utilities peak at 25, 50 agents' utilities peak at 50, and 25 agents' utilities peak at 75, with $\sigma_i = 50$ for all agents i, total utility is maximized if the 50 agents for which $\mu_i = 50$ go to the bar. Can the mayor elicit this efficient outcome?

Let us begin our analysis by giving the mayor access to an oracle that informs him of all the agents' peak valuations. In this case, he can compute the median, which is 50 in our example. But can he induce precisely those 50 agents whose peak utilities occur at 50 to attend the bar so as to maximize total utility? The following taxation scheme is designed to achieve this outcome: charge agents that attend the bar an entrance fee, say x, and distribute the proceeds evenly among those agents that do not attend the bar. Suppose λ agents go to the bar. The payoffs to those λ agents are $u_i(\lambda) - x$, if the bar is uncrowded, and $-(u_i(\lambda) - x)$, if the bar is crowded. The payoffs to those $N - \lambda$ agents that do not go to the bar are $\lambda x/(N - \lambda)$. An entrance fee $x = 0.5$ yields an equilibrium in which the 50 agents whose peak valuations are 50 attend the bar, and the remaining 50 agents stay home. All agents achieve utility 0.5 in equilibrium. Allowing Q-learning agents [9] to adapt their behavior to this mechanism, average utility per agent approaches this equilibrium value.

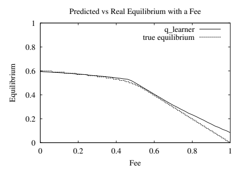

Figure 9: The predicted equilibrium verse the fee

See Figure 1(a).

But in fact the mayor does not have access to an oracle that informs him of all the agents' peak valuations. Thus, he cannot compute the median, and he cannot compute the equilibrating fee. His behavior, too, is constrained to be boundedly rational. As both *planner* and *player*, assume the mayor sets a fee according to his beliefs about the population of Santa Fe, and updates the fee weekly according to the trend in utility. One simple algorithm for adjusting the fee is *derivative-following* [5], which experiments with incremental changes, continuing to move in the same direction until average utility decreases, at which point the direction of movement is reversed: given increment $\beta > 0$, the fee $f_{t+1} = f_t + \beta[\text{sign}(f_t - f_{t-1})\text{sign}(u_t - u_{t-1})]$, where u_t is the agents' average utility at time t. Figure 1(b) depicts simulations not only of boundedly rational agents adapting their behavior to the mechanism, but the planner adjusts the mechanism in response to the agents' collective choices as well. Both the fee and average utilities per agent converge near the equilibrium value of 0.5.

Now suppose the agents' valuation functions are such that $\mu_i = \sigma_i = 60$ for all i. In this setting, it is appropriate to define a social choice not only as an efficient outcome (*i.e.*, utility-maximizing), but among all efficient outcomes, a social choice should be *fair*. Fairness requires that agents with identical preferences be equally likely to attend the bar. Note that the mixed strategy Nash equilibrium satisfies fairness, but it is inefficient—it yields an average utility near zero—since half the time the bar is overcrowded. Boundedly Rational Q-learning agents together with a planner that abides by the derivative-following algorithm generate fair outcomes in this scenario that approximate efficiency. See Figure 1(c). This taxation scheme is an example of an iterative mechanism for which a set of boundedly rational agents together with a boundedly rational planner approach equilibrium behavior that implements an approximate social choice.

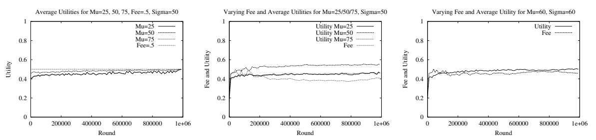

Figure 10: (a) A heterogeneous population of Q-learners learn the equilibrium of the mechanism: *i.e.*, the social choice. (b) A heterogeneous population of Q-learners learn the equilibrium of the mechanism while a derivative follower simultaneously implements an efficient social choice. (c) A homogeneous population of Q-learners learn the equilibrium of the mechanism while a derivative follower simultaneously implements a fair and efficient social choice.

8 Conclusions

Our summer of Santa Fe Bar hopping carried a number of exceptional results with it. Though Hedge did not reveal any substantial findings, both Q learning and an added fee did. Infact, all of the agents trained as either discrete three state Q learners or continuous two state q learners with the added fee converged to having an average per game utility of at least .5.

Considering that this is almost 85% of the maximum possible per game utility, we are very pleased with the results.

Future research about the SFBP will hopefully both improve upon the .5 average per game utility and explore more varied bar setups and scenarios.

References

[1] Jutsin Boyan. Learning evaluation functions for global optimizations. *PHD Thesis*, page 216, August 1, 1998.

[2] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Computational Learning Theory: Proceedings of the Second European Conference*, pages 23–37. Springer-Verlag, 1995.

[3] A. Greenwald. *Learning to Play Network Games*. PhD thesis, Courant Institute of Mathematical Sciences, New York University, New York, May 1999.

[4] A. Greenwald, A. Jafari, G. Ercal, and D. Gondek. On no-regret learning, Nash equilibrium, and fictitious play. In *Proceedings of Eighteenth International Conference on Machine Learning*, pages 226–233, June 2001.

[5] A.R. Greenwald and J.O. Kephart. Shopbots and pricebots. In *Proceedings of Sixteenth International Joint Conference on Artificial Intelligence*, volume 1, pages 506–511, August 1999.

[6] R.B. Myerson. *Game Theory: Analysis of Conflict*. Harvard University Press, Cambridge, 1991.

[7] M. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, Cambridge, 1994.

[8] T. Schelling. *Micromotives and Macrobehavior*. W.W. Norton and Company, New York, 1978.

[9] C. Watkins. *Learning from Delayed Rewards*. PhD thesis, Cambridge University, Cambridge, 1989.