WORKING PAPER No.210

Multi-Agent Learning in Dynamic Pricing Games of Service Markets

by

Diatha Krishna Sundar
&
K. Ravikumar

July 2003

Please address all correspondence to:

Prof. D. Krishna Sundar
Indian Institute of Management Bangalore
Bannerghatta Road
Bangalore – 560076, India
E-mail: diatha@iimb.ernet.in
Phone : 080 – 699 3276
Fax : 080 - 6584050

K. Ravikumar
IBM India Research Laboratory
Indian Institute of Technology
New Delhi 110 016, India
E-mail : rkaruman@in.ibm.com

Abstract

We study price dynamics in a service market environment where identical service providers dynamically reset their prices to price discriminate *informed* and *uninformed* consumers. A semi-Markovian game model for dynamic pricing is developed and a new *multi-time scale actor-critic* algorithm is proposed for multi-agent reinforcement learning. Also, experimental results on convergence to a Nash equilibrium are presented.

1 Introduction

E-commerce has undoubtedly changed how business is done. On the Internet, competition is just a click away. This fact has potentially lead to intense price competition for commodity products. Search engines like ACSES, and Web-based comparison shopping agents (also known as *shopbots*), like Dealpilot.com allow consumers easy access to all competing firms' prices. In order to attract consumers, sellers use automated pricing agents, (also called *pricebots*) for constant resetting of prices. Developing on Varian's work (Varian 1980) in a physical retailer market, Greenwald, Kephart and Tesauro (1999) have investigated "economics of shopbots" and pricebot dynamics in commodity markets with infinite supply. In their models, some consumers have access to shopbots while other consumers do not. These models generate equilibria: firms randomize their prices in order to price discriminate between the searchers and non-searchers.

In the same vein, in this paper, we study price dynamics in an electronic service market with sellers of identical service. This models a situation where online utility services or digital goods or videos are offered for rent. Also, Application Service Providers (ASPs) form a good example of the model presented here.

Since a seller of a service can process requests of consumers only at a finite rate, a buyer approaching for service will incur waiting cost before his request is initiated. In our model, we assume that shopbots not only will collate *posted prices* of all the sellers but also provide information pertaining to *posted expected delay* at each such service provider. Further, each service provider uses an automated pricing agent (or pricebot) to reset prices at random intervals.

Congestion is a characteristic of services and creates *negative network externalities*: the utility to each consumer decreases with an increase in the total number of purchasers. In

a market without congestion, competition in prices drives them downward resulting in zero profits. In contrast, in service markets, the downward pressures on prices is countered to some extent by the congestion or *dis-utilities* consumers incur from waiting costs. Competing firms might choose to differentiate themselves by offering different prices, and thus different qualities of service in terms of congestion. In a duopoly setting, one firm will offer a higher price than the other, appealing only to the most congestion sensitive customers while the majority of customers will use the service of the other, less expensive firm. If the customers approaching the market are heterogenous in their information acquisition capabilities or in their preferences for sellers, competing firms have incentive to randomize their prices in order to price discriminate such customers: a phenomenon widely prevalent in commodity markets. In this paper, we attempt to make this intuition precise by developing a truly dynamic game model in a duopoly setting and study its dynamics when pricing agents dynamically adjust prices based on competitor's behavior.

Nash equilibrium is a natural equilibrium concept in such games and is a point in the joint policy space where no seller has incentive to deviate unilaterally. If all the sellers follow the same rational learning algorithm that consistently attempts to learn a seller's best response to opponents' actions, and if it *converges*, then the sellers will be *locked in* such a Nash equilibrium.

In this paper, we analyze price dynamics in a competitive two-seller market game where both the players follow the *reinforcement learning* based adaptive behavior. Q-learning based algorithms have been suggested in Littman (1994, 2001)and Hu and Wellman (1998). However, satisfactory convergence is confined only to zero-sum games where one person's pay-off is negative of the other player's pay-off. In contrast, we consider a general-sum Markovian game and propose

an actor-critic-type of reinforcement learning scheme, a variant of the type discussed in Konda and Borkar (2001) and Borkar (2002) in the following sense: we model the two players as two actor-critic learners, but the actors (policies) are updated on different time scales with the intuition that if two actors run on different time scales, the slower player sees the other player as "equilibrated" and the faster player sees the other player as quasi-static and hence, both the learners might converge (to a Nash equilibrium). This is a reasonable model when the competing sellers differ in their technologies and information acquisition capabilities. We provide experimental results on convergence in a dynamic pricing game. Though no claims can be made on convergence of the algorithm in general, this work can be treated as a first step in that direction.

In the next section we introduce the dynamic pricing model in service markets. In Section 3 we model the dynamic pricing game and present our multi-time scale actor-critic algorithm. Section 4 gives results of our experimentation.

2 Description of the Model

We consider a simple model of a service market with two service providers. A Poisson stream of buyers with rate λ approaches the market with $i.i.d$ service time requirements sampled from exponential distribution having mean $\frac{1}{\mu}(\lambda < \mu)$. Buyers are classified into two categories: A Type 1 buyer randomly chooses a service provider and receives a quote on price to render requested service and the expected delay to be incurred to initiate processing his request. In contrast, Type 2 buyers use a *shopbot*, to learn *posted price* quote and also the *posted expected delay* at each individual service provider. In both the cases, a seller is assumed to act truthfully

in revealing information regarding delay quotes and with n requests queued up, will quote a delay of $\frac{n}{\mu}$. Though it is interesting to analyze the game where the above assumption is relaxed, we do not take up that part in this paper.

Each seller can process only one request at a time and seller l, $l = 1, 2$, has a finite buffer to allow at most m_l requests at any time. Further, a seller uses his own automated price-setting agent, a *pricebot*, to price the requested service dynamically based on competitive factors, current queuelength and also, based on relative proportion of the informed buyers (Type 2 above) approaching the market.

Now, consider a customer of type m, $m = 1, 2$ arriving at time t. Let $X_l(t)$ be the number of requests in queue at server l, $l = 1, 2$ and $W_l(t)(= \frac{X_l(t)}{\mu})$ be the corresponding expected delay. Let $\mathbf{X}(t) = [X_1(t), X_2(t)]$ be the queuelength vector. $\mathbf{X}(t-)$ $(\mathbf{W}(t-))$ will denote the queue vector (expected delay vector) 'just before t'. The utility for the customer of price and delay quotes received at seller l is given by $\psi_l(p_l, W_l(t-), m)$ for some map ψ_l. Further, a customer of type 1 selects seller l with probability α_l and joins the queue for service if $\psi(.)$ is positive or else leaves the market. But a customer of type 2 joins the k-th seller, if k maximizes $\psi_l(p_l, W_l(t-), 2)$ $l = 1, 2$ and $\psi_k(.)$ is positive.

2.1 The Semi-Markovian Game

We assume that each seller can have knowledge of *changes* in the queuelengths at the other seller. It is not a restrictive assumption since sellers always can approach shopbots for this information. However, it is interesting to note that shopbots will not have any incentive to reveal price information to sellers since a portion of shopbot's revenue comes from purchase

deals that materialize through it and no such deal is possible if the query on price comes from a seller. Since this is an assumption on observability of only the changes in the competitor's queuelengths, information with regard to lost arrivals and type of customers is not explicitly available to any seller.

Now we are ready to formulate a simultaneous move dynamic game between the two sellers. In the following, we use Player 1 and seller synonymously and refer to his/her opponent as competitor or Player 2.

Define $S = \{0, 1, 2, \ldots, m_1\} \times \{0, 1, 2, \ldots, m_2\}$ where m_1 and m_2 are the buffer sizes at the seller and the competitor respectively. Let its elements be enumerated as $1, 2, \ldots, M$.

Consider the process $\{\mathbf{X(t)}\} = [X_1(t), X_2(t)], t \geq 0$ of queuelength vectors controlled by the pricing strategies of the two players as follows:

At time 0, the process is observed and classified into one of the states in S. After identification of the state, the players choose pricing actions from a finite set of pricing actions which for notational convenience is assumed to be common for the two players and is denoted by $\mathcal{A}$. If the process is in state i and Players 1 and Player 2 choose a_1 and a_2 respectively, where $a_l \in \mathcal{A}$, then

(i) the process transitions into state $j \in S$ with probability $P_{ij}([a_1, a_2])$

(ii) and further, conditional on the event that the next state is j, the time until transition is a random variable with probability distribution $F_{ij}(.|[a_1, a_2])$.

After the transition occurs, pricing actions are chosen again by the players and (i) and (ii) are repeated.

Further, in state i, the actions chosen by players 1 and 2 are a_1 and a_2 respectively, and the process moves to state j, the resulting rewards are given as follows:

Let $i := [x_1, x_2]$ and $j = [x_1', x_2']$.

$$\begin{aligned} r^l(i,j;\mathbf{a}) &= 0 \text{ if } x_l = x_l' \text{ or } x_l' = x_l - 1 \\ r^l(i,j;\mathbf{a}) &= a_l \text{ if } x_l' = x_l + 1 \\ c^l(i) &= c^l x_l \end{aligned} \tag{1}$$

Following along the lines of semi-Markov decision processes, we call the above controlled game process a *semi-Markov game*. From standard semi-Markov decision theoretic arguments it follows that we may consider the embedded discrete time Markov game $\mathbf{X}(\tau_n)$ controlled by the two players at the transition epochs τ_n. At every such τ_n, if the action profile, the ordered pair of pricing strategies by Player 1 and Player 2, is $\mathbf{a}^n = [a_1^n, a_2^n]$ with $a_l^n \in \mathcal{A}$, then there is a potential departure from the queue of seller l, with probability $\frac{\mu}{\lambda+2\mu}$. With probability $\frac{\omega_2\lambda}{\lambda+2\mu}$, there is an arrival of type 2, who joins the queue of seller k if k maximizes $\psi_l(X_l(\tau_n-), a_1^n, 2)$, if ψ_k is positive and if k-th buffer is not full or leaves the system. Similarly, with probability $\frac{\omega_1 a_l \lambda}{\lambda+2\mu}$, the arrival of type 1 joins the queue of seller l again only when $\psi_l(X_l(\tau_n-), a_l^n, 1)$ is positive.

Let $\{\tau_n\}_{n\geq 0}$ denote the sequence of successive transition epochs. We use $\mathbf{X}_k$ to denote $\mathbf{X}(\tau_k)$ and $\mathbf{a}_k$ to denote $\mathbf{a}_{\tau_k}$. Given at a decision epoch τ_n, the state of the game and the strategy profile $\mathbf{a}_n$, the joint distribution, $Q_{ij}(t, \mathbf{a})$, of the transition interval and the next state,

is by time homogeneity:

$$\begin{aligned} Q_{ij}(t,\mathbf{a}) &= P\{\tau_{n+1}-\tau_n \le t, X_{k+1}=j | X_k=i, \mathbf{a}_k\} \\ &= P_{ij}(\mathbf{a})F_{ij}(t|\mathbf{a}). \end{aligned} \tag{2}$$

Now consider the control processes $\{Z_n^l\}$ of the sellers at decision epoch τ_n taking values in the pricing action space $\mathcal{A}$. Let $Z_n = [Z_n^1, Z_n^2] \in \mathcal{A}$ be the joint process. Then, transitions of the game occur as specified in (i) and (ii) above. Player l seeks to maximize her total expected discounted reward over the infinite horizon of the game:

$$\begin{aligned} & E[\sum_{n=0}^{n=\infty} e^{-\beta\tau_{n-1}}(r^l(X(\tau_{n-}), Z_n, X(\tau_n)) \\ - & \int_{\tau_{n-1}}^{\tau_n} c^l(X(\tau_{n-1})e^{-\beta t}dt) | X_1 = i \ \ \forall i \in S] \end{aligned}$$

$\beta \in (0,1)$ above is a discount factor that discounts all future rewards to the initial decision epochs.

A policy π^l for player l, $l = 1,2$ is a sequence $[\pi_1^l, \pi_2^l, \ldots]$ where π_n^l specifies the action to be chosen at the n-th decision epoch, which in general can be dependent on the entire history up to τ_n. A policy for player l, π^l is called *stationary* if there is a map $f^l : S \to \mathcal{P}(\mathcal{A})$ where $\mathcal{P}(\mathcal{A})$ is the class of probability distributions over the space of pricing actions $\mathcal{A}$. Because such a policy is time and history independent, can simply be denoted by π^l. Further, a stationary policy profile π, is the ordered pair $[\pi^1, \pi^2]$ of stationary policies $\pi^l, l = 1,2$ and $\pi \in \mathcal{P}(\mathcal{A})^{2M}$.

In order to help write a recursive scheme to evaluate total expected reward for a given π and thus to motivate our learning scheme to be described later, we define the following single stage

terms:

$$m_{ij}(\pi) := \sum_{a_1,a_2} \int_0^\infty e^{-\beta\tau} \pi^1(i,a_1)\pi^2(i,a_2) dQ_{ij}(t,\mathbf{a})$$

$$\overline{r}^l(i,\pi) := \sum_{j=1}^{M} \sum_{a_1,a_2} P_{ij}(\mathbf{a}) r^l(i,j,\mathbf{a}) \pi^1(i,a_1)\pi^2(i,a_2)$$

$$\overline{G}^l(i,\pi) := c^l(i) \sum_{a_1,a_2} \sum_{j=1}^{M} \int_0^\infty \frac{1-e^{-\beta\tau}}{\beta} dQ_{ij}(t,\mathbf{a})$$

Note that transitions to states and transition intervals are governed by both the players and depend on the randomized actions $\pi^l(.,.)$ taken by both the players and also on buyer's utilities. $m_{ij}(\pi)$ above is the expected discount factor if the state of the game next transitions to j when in state i players follow π. $\overline{r}^l(i,\pi)$ and $\overline{G}^l(i,\pi)$ are the expected reward and the expected costs for a single transition if the game starts in state i and the pricing action selected by the players is according to π. When π is a profile of pure (deterministic) strategies with unit mass on $[a_1,a_2]$, we denote $m_{ij}(\pi)$ by $m_{ij}([a_1,a_2])$ and $\overline{G}^l(i,.)$ above by $G^l(i,[a_1,a_2])$.

For a stationary policy profile π, define the constant policy evaluation function" $\overline{V}^l_\pi$ for player l by

$$\begin{aligned} \overline{V}^l_\pi(i) = \; & E[\textstyle\sum_{n=0}^{\infty} e^{-\beta\tau_{n-1}} (\overline{r}^l(X(\tau_{n-}), Z_n, X(\tau_n)) \\ & - \textstyle\int_{\tau n-1}^{\tau_n} c^l(X_n) e^{-\beta t}\, dt) | X_0 = i] \forall i \in S \end{aligned} \tag{3}$$

where $\{Z_n\}$ are chosen according to the stationary (randomized) policy π. Following standard semi-Markov decision theoretic arguments it is easy to show that $\overline{V}^l_\pi(.), l = 1,2$ is the unique

solution to the fixed point equation:

$$\overline{V}_{\pi}^{l}(i) = \overline{r}^{l}(i,\pi) - \overline{G}^{l}(i,\pi) + \sum_{j} m_{ij}(\pi)\overline{V}_{\pi}^{l}(j) \forall i \in S \tag{4}$$

We call the policy profile $\pi(.,.)$ a *Nash equilibrium* if for every l, $V_{\bar{\pi}}^{l}(i) \leq V_{\pi}^{l}(i) \quad \forall i$ whenever, $\bar{\pi}^{k}(.,.) = \pi^{k}(.,.)$, for $k \neq l$.

Such a Nash equilibrium can be shown to exist following the arguments in Federgruen (1978) for discrete games. Moreover, if we freeze policies for one agent, it becomes a semi-Markov Decision Process for the other agent whence it follows that $V_{\pi^2}^{k}(i)$ satisfies the following dynamic programming equation: $\forall x \in S$,

$$\begin{aligned}\overline{V}_{\pi^2}^{1}(i) = \min_{a} \sum_{a_2} \pi^{2}(i,a_2)[P_{ij}([a,a_2])r^{1}(i,[a,a_2],j) \\ -G(i,[a,a_2]) + \sum_{j \in S} m_{ij}([a,a_2])\overline{V}_{\pi^2}^{l}(j)]\end{aligned} \tag{5}$$

Similar relation holds for $\overline{V}_{\pi}^{2}(.)$. In particular, it also follows that (I) π^{l} itself is supported on the *argmin* of the r.h.s above and (II) Player l cannot do any better by using any other general non-anticipative policy.

3 An Actor-Critic Type of Algorithm for the Pricing Game

From the remarks below equation (5), it is enough for the players to concentrate on stationary policies to play the game. Now assume that a player, say Player 2, follows a fixed stationary policy and further that, the policy is known to Player 1. For the sake of argument, let us also

assume that buyers' behavior and their utilities are also common knowledge. In other words, all the parameters of the game are known. Then, Player 1 has to solve (5) to find his best response to Payer 2's strategy.

Consider the celebrated policy iteration scheme for solution of (5) which is detailed below:

Player 1 starts with a guess for optimal stationary deterministic policy $\pi^0(.)$ and at iteration $n \geq 0$ does the following:

(a) Find $\overline{V}^n_{\pi^2} : S \rightarrow \mathcal{A}$ by solving

$$\overline{V}^n_{\pi^2}(i) = \sum_{a_2} \pi^2(i,a_2)[P_{ij}([\pi^0(i),a_2])r^1(i,[\pi^0(i),a_2],j) \\ -G(i,[\pi^0(i),a_2]) + \sum_{j \in S} m_{ij}([\pi^0(i),a_2])\overline{V}^n_{\pi^2}(j)]$$

(b) Set $\pi^{n+1}(i)$ as an element in

$$Argmin(\sum_{a_2} \pi^2(i,a_2)[P_{ij}([.,a_2])r^1(i,[.,a_2],j) \\ -G(i,[.,a_2]) + \sum_{j \in S} m_{ij}([.,a_2])\overline{V}^n_{\pi^2}(j)]$$

Now let us relax the earlier assumption on common knowledge about buyers' behavior. In this case, the transition structure above is not known and one has to use adaptive mechanisms based on reinforcement learning. We develop an actor-critic type of reinforcement learning for the above game similar in spirit to the one in Konda and Borkar (1998). To motivate the algorithm, replace the step (a) above, by the following iterative scheme to solve the underlying linear system of equations.

(**a'**) $\overline{V}^n_{m+1}(i) =$

$$\sum_{a_2} \pi^2(i,a_2)[P_{ij}([\pi^0(i),a_2])r^1(i,[\pi^0(i),a_2],j)$$
$$-G(i,[\pi^0(i),a_2]) + \sum_{j\in S} m_{ij}([\pi^0(i),a_2])\overline{V}^n_m(j)] \tag{6}$$

Note that this can be considered as a subroutine to perform the task of step (**a**). If the transition structure is not known, then the conditional averaging in (6) cannot be performed. One might then consider replacing the conditional average in (6) by an actual evaluation at states and transition intervals obtained from online learning. In other words,

$$V^n_{m+1}(i) = V^n_{m+1}(i) + b^l(\upsilon(i,m))I_{\{X_m=i\}}[(r^l(i,Z_m) -$$
$$\frac{1-e^{\beta\tau}}{\beta}c(i,Z_m) + e^{-\beta\tau}V^n_{m+1}(X_{m+1}) - V^n_{m+1}(i)]$$

Now consider (**b**), the policy improvement step, of the policy iteration, which entails solving an optimization problem for each iteration n. In order to do so, it needs to wait for the policy evaluation step to converge and then reevaluate the new policy again following the above learning procedure. To obviate this difficulty, one may try to execute both the steps together for online learning and update policies and values in a *coupled* fashion : the policy is updated using an approximate gradient scheme (to be detailed shortly). The gradient estimate is derived from the available estimates of the value function obtained from the above learning step. But importantly, to underscore the fact that policy update should wait till convergence of step (**a'**), the policy update is run at a slower time scale than the value update, that is, step (**a'**) to get the same effect albeit asymptotically. This notion is formalized later.

Now the above argument assumes that Player 2 is follows a fixed stationary strategy. However, if Player 2 were also to learn his best strategy, and hence both hope to head to Nash equilibrium, then simultaneous adaptation of both the players creates convergence problem. In this case, where both the agents try to learn their Nash equilibrium strategies following best response dynamics, it can be hoped that both will converge to an equilibrium (more so, if it is unique) if Player 1 *sees* Player 2 as quasi-static and Player 2 *sees* Player 1 as playing equilibrium strategy in their pursuit for mutual best responses. With this intuition, we devise two similar actor-critic learners that operate on different time scales for updates. This notion will be made precise shortly.

As it is known that existence of equilibrium for the above dynamic pricing game can be ascertained only in the randomized policies (perhaps not in pure strategies), let us extend the domain of optimization in (**b**) to the space of probability measures on action space. Advantageously, this space is convex and hence one can use gradient based numerical schemes to solve the underlying optimization problem at each step of learning. Approximate gradient estimate is provided through step (**a'**). Whenever this updated policy falls out of the boundary of the convex policy space, it is projected back to the convex domain. This procedure is formalized below.

Consider the simplex of probability vectors over the action space A, $\mathbf{P}(A)$. Any stationary randomized policy for Player l is a map $\pi^l : S \to \mathbf{P}(A)$. For $i \in S$, $\pi^l(i)$ is an $|A|$- vector whose components are denoted by $\pi^l(i,a), a \in A$. We search for optimal $[\pi^l(i,a)]_{i \in s, a \in A}$ in $(\mathbf{P}(A))^M$.

The actor-critic algorithm for player, l, $l = 1, 2$, is defined as follows. Equations (4) and (5) suggest the following update procedures for the critic (*policy evaluator*) and the actor (*policy*)

respectively.

For any $i \in S$,

$$V^l_{n+1}(i) = V^l_n(i) + b^l(v(i,n))I_{\{X_n=i\}}[(r^l(i,Z_n) - \frac{1-e^{\beta\tau}}{\beta}c(i,Z_n) + e^{-\beta\tau}V^l_n(X_{n+1}) - V^l_n(i)] \tag{7}$$

$$\hat{\pi}^l_{n+1}(i.,) =$$

$$\Gamma(\hat{\pi}^l_n(i,.) + \sum_{a\neq a_0} a^l(\nu(i,a,n))I_{\{X_n=i,Z^l_n=a\}}(r^l(i,Z_n) - \frac{1-e^{\beta\tau}}{\beta}c(i,Z_n) + e^{-\beta\tau}V^l_n(X_{n+1}) - V^l_n(i))e_a) \tag{8}$$

where e_a is the unit vector with value 1 in the $a-th$ position,$\{a^l(n)\}$ and $\{b^l(n)\}$ are the step size parameter sequences satisfying the standard stochastic approximation conditions and $\nu(i,a,n)$ is the number of times (i,a) is encountered in the chain $\{(X_n,Z_n)\}$ and and $v(i,n)$ is the number of times state i is visited by time n. $\Gamma(.)$ is the projection on to the probability simplex $P_0(A) := \{x : \sum_i x_i = 1, x_i \geq 0, \forall i\}$. Finally, let $\varepsilon \in (0,1)$ be a small positive number. Then, player l picks Z^l_n according to the distribution $\pi^{l\,\varepsilon}_n(X_n,.)$ defined for any $\phi \in (\mathbf{P}(A))^M$, by $\phi^\varepsilon(i,.) := \varepsilon\varsigma + (1-\varepsilon)\phi(i,.)$ where ς is the uniform distribution over A to ensure sufficient exploration. τ above is an appropriately averaged sample transition time.

In addition to the standard conditions on $\{a^l(n)\}$ and $\{b^l(n)\}$ for stochastic approximation

schemes, we also require that the sequences $\{a^l(n)\}$ and $\{b^l(n)\}$ satisfy:

$$a^l(n) \quad = o(b^l(n)), i = 1, 2 \quad \text{and} \quad a^1(n) = o(a^2(n)) \tag{9}$$

If one interprets $\{a^l(n)\}$ and $\{b^l(n)\}$ as time scales, then (9) defines three time scales for operation of the two actor-critics; while the two actors operate on different time scales, their respective critics operate on the same time scale faster than their respective actors.

3.1 Computing Projection

In this section, we provide a simple algorithm to compute the above projection under L^2-norm to the probability simplex.

Given any vector $\mathbf{a} = [a_1, a_2, \ldots, a_m]$, finding its projection on the probability simplex, $\mathbf{P}(A)$ under the L^2 norm amounts to solving

$$\begin{aligned} &\min \textstyle\sum_{i=1}^{m} (x_i - a_i)^2 \\ &s.t \ \ \textstyle\sum_{i=1}^{m} x_i = 1 \\ &x_i \geq 0 \ \ \forall i \end{aligned} \tag{10}$$

Without loss of generality, we may assume that $a_1 \geq a_2 \geq \ldots \geq a_m$.Let x^* denote the optimal solution to the above problem. Then

Lemma 1 *There exists an index k such that $x_k^* > 0$, for $1 \leq i \leq k$ and $x_j^* = 0$, for $i > m$.*

Proof: Suppose that x^* is such that $x_i^* = 0$ and $x_{i+1}^* > 0$ for some i and let $d(x^*$ be its distance

from the $\mathbf{P}(A)$. Then consider a solution $\tilde{x}$, obtained from x^* such by switching the i^{th} and the $i+1$-th components. The distance of $\tilde{x}$, $d(\tilde{x})$ from $\mathbf{P}(A)$ is

$d(\tilde{x} = d(x^*) + 2(a_{i+1} - a_i)x_{i+1}$

Since, $a_i \geq a_{i+1}$, x^* cannot be the projection. Δ

Lemma 2 *All positive components of x^* are of the form $x_i^* = a_i + c$ for some constant c.*

Proof: The lemma is trivial if there is only one component that is positive. Hence, assume that there are two positive elements, say x_i^* and x_j^*, in x^*. Define a new feasible solution, $\tilde{x}$, as follows.

$\tilde{x}_i = x_i^* - \epsilon$ and $\tilde{x}_j = x_j^* + \epsilon$, and $\tilde{x}_l = x_l^*$ for $l \neq i, j$ for some $\epsilon > 0$.

The distance of $\tilde{x}$ to $\mathbf{P}(A)$ is

$d(\tilde{x}) = d(x) + 2\epsilon^2 + 2\epsilon(x_j^* - x_i^* - a_j + a_i)$

For ϵ sufficiently small, $\tilde{x}$ is a feasible solution and further, if $(x_j^* - x_i^* - a_j + a_i) \neq 0$, then $\tilde{x}$ has lesser distance than x^* from the simplex, contradicting the optimality of x^*. Δ

In view of the characterizations for x^* in the above lemmata, the optimization problem reduces to a problem in single variable:

$$\begin{aligned} &\min \tfrac{1}{k}\left(1 - \textstyle\sum_{l=1}^{k} a_l\right)^2 + \textstyle\sum_{l=k+1}^{m} a_l^2 \\ &s.t \ \ \textstyle\sum_{l=1}^{k}(a_l - a_k) \leq 1 \\ &k \in 1, 2, \ldots, m \end{aligned}$$

Now, consider the sequence, $S_k = \sum_{l=1}^{k}(a_l - a_k)$ with $S_1 = 0$. It is easy to see that $\{S_k\}$ is

non-decreasing. The objective function can be rewritten in a recursive form as follows: $f(k) = f(k-1) - \frac{1}{m(m-1)}[1 - S_k]^2$ with $f(1) = 1 - 2a_1 + \sum_{l=1}^{m} a_l^2$. It can be easily verified that $f(k)$ is *non-increasing.*

In the light of these facts, the problem reduces to determining k^*, the largest k such that $S_k \leq 1$. Such a k^* exists since $S_1 = 0$.

The optimal value, c^*, in terms of k^* is $c^* = \frac{(1-S_{k^*})}{k^*} - a_{k^*}$ and the projection is given by

$$x_i^* = \begin{cases} a_i + c^* & \text{for } i = 1, 2, .., k \\ 0 & i > k^*. \end{cases}$$

4 Simulation Study

In this section, we provide some sample experimental results on convergence behavior of the algorithm. We consider two cases: in the first case, each seller is simulated with finite processing capacity and finite buffer, and in the other case, the processing capacity of each seller is assumed to be infinite.

Each buyer has his own upper-limits, p_b, w_b, on price and waiting time respectively and are assumed to be $i.i.d$ and uniformly distributed over the intervals $(0, p_{max}]$ and $(0, w_{max}]$ respectively for known p_{max} and w_{max}. A typical buyer's utility is assumed to be of the following form:

$$U_b(p, w) = [\alpha(p_b - p) + (1 - \alpha)(w_b - w)]\Theta(p_b - p)\Theta(w_b - w) \tag{11}$$

where $\Theta(x) = 1$ if $x \geq 0$ and $\Theta(x) = 0$ otherwise, for any quoted price p and waiting time w and $0 \leq \alpha \leq 1$.

4.1 Sellers with Finite Capacity

In this case, with a view to get immediate insights that help analyze the convergence behavior, the dynamic pricing game has been simulated in a completely symmetric setting wherein the sellers are assumed to have identical buffer capacity of 4 and identical holding cost of 1 and choose their pricing actions from the set $\{1, 2, 3, 4, 5, 6\}$. The game has been simulated in two different scenarios: one with a mix of informed and uninformed buyers and the other with only informed buyers. In the latter case, buyers' price barriers and waiting time barriers are set at 6 and 5 respectively and a buyer's priority (α) to surplus extracted from price is sampled from a uniform distribution over $(0, 1]$. In the former scenario, the price and waiting time barriers are sampled from uniforms distributions over $(0, 6]$ and $[0, 5]$ respectively keeping the value of α fixed at $\frac{1}{2}$. The learning rate parameters are of the following form: $a^1(n) = \frac{1}{n}; a^2(n) = \frac{1}{n^{0.75}}; b^1(n) = \frac{1}{n^{0.6}}$ and $b^2(n) = \frac{1}{n^{0.6}+100}$. The system has been simulated for 5 million iterations. The convergence is slow which is typical of reinforcement learning algorithms. The plots shown below are the values in any given state against the number of updates in the last 15000 iterations.

Plots on convergence of value functions when system starts from the idle state in the above scenarios have been presented in Figure 1 and Figure 2 respectively. Observe that the rewards obtained in the presence of only informed buyers are lower compared to the situation where a mix of consumers is present. The convergence in Figure 2 is not as smooth as that in Figure 1. However, it has been observed that values differ in their fourth decimal and show slight perturbations around 0.018. For purposes of visual clarity, we have not provided convergence graphs of mixed strategies in each state. In the second scenario, for state (0,0), the policies of the two players converge approximately to the mixed strategy [0.013, 0.012, 0.230, 0.235,

0.240, 0.270]. The congestion factor provides enough incentive for the sellers to randomize on prices above the minimal level. Contrast this with a commodity market (with no constraints on supply) where informed buyers drive the prices down to the lowest possible price (or marginal cost), the popular *Bertrand equilibrium.*

Figure 3 depicts value function convergence starting from (0,1).

For state (3,1), (no plots are provided here in this case) strategies for Seller 1 and Seller 2 converge to [0.07, 0.08, 0.11, 0.13, 0.27, 0.33], [0.040, 0.044, 0.099, 0.200, 0.297, 0.320]. It is interesting to note that in this state both the sellers randomize their prices in high price domains. This phenomenon can be explained as follows: while Seller 1 tries to reduce congestion by discouraging price sensitive customers with high price quotes, the competitor tries to derive advantage out of the situation: the congestion sensitive consumers select him out of priority and price sensitive consumers out of better service offer at similar price quotes.

4.2 Sellers with Infinite Capacity

We also simulated the case where both the sellers can process requests at an infinite rate. In other words, each seller can be treated as an infinite server Markovian queue. This models a situation when servers have enough resources to meet the demand. The mixed strategy obtained after convergence for the above price grid is [0.5,0.0,0.0,0.0,0.0,0.5] for Player 1 and [0.0, 0.0, 0.55, 0.45, 0.0, 0.0] for Player 2. Figure 4 is a combined plot for both the players: Action 6 for Player 1 and Action 3 for Player 2.

In another experiment, we selected only two prices and the convergence in policy is depicted in Figure 5.

In the third experiment, we assumed that all customers are informed and no customer balks. That is, a customer joins the queue that offers maximal utility even if the utility is negative. Hence in this case, the game is a zero sum Markovian game and the game reduces to a linear dynamical system. Again the policies converge to a mixed strategy as in the finite capacity case: congestion factor counters the effect of competition and prevents the optimal price from going down to marginal cost. The plot shows convergence graph of probability of selecting price 2. The policies again turned out to be symmetric in this case.

5 Conclusions

In this paper, we developed a multi-time scale algorithm and provided empirical results on convergence in a truly dynamic game. Our studies indicate that multi-time scale algorithms offer some promise in multi-agent learning in games.

References

[1] Barto, A., Sutton, R., & Anderson, C. (1983). Neuron-like elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man and Cybernetics*, **13**: 835-857.

[2] Borkar, V. S. (In Press). Reinforcement learning in Markovian evolutionary games. *Advances in Complex Systems.*

[3] Federgruen, A. (1978). On N-person stochastic games with denumerable state space. *Advances in Applied Probability*, **10**: 452-471.

[4] Greenwald A. R., Kephart J. O., & Tesauro G. J. (1999). Strategic pricebot dynamics. *Proceedings of the First ACM Conference on E-Commerce*, pp. 58-67. Morgan-Kaufmann.

[5] Hu, J.C., & Wellman, M. (1998). Multi-agent reinforcement learning: theoretical framework and an algorithm. *Proceedings of the Fifteenth International Conference on Machine Learning*, pp. 242-250. Morgan-Kaufmann.

[6] Konda V. R., & Borkar V. S. (1999). Actor-critic type learning algorithms for Markov decision processes. *SIAM Journal on Control and Optimization*, **38**: 94-123.

[7] Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. *Proceedings of the Eleventh International Conference on Machine Learning*, pp. 157-163. Morgan-Kaufmann.

[8] Littman, L. M. (2001). Friend-or-foe Q-Learning in general-sum games. *Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 322-328. Morgan-Kaufmann.

[9] Varian, H. R. (1980). A model of sales. *American Economic Review*, **70**: 651-659.

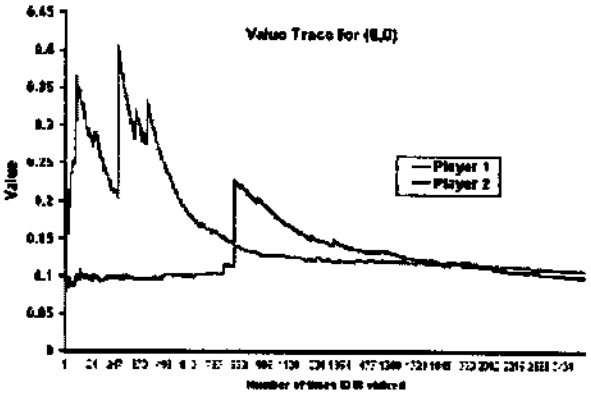

Figure 1: V(0,0) with Mix of Buyers

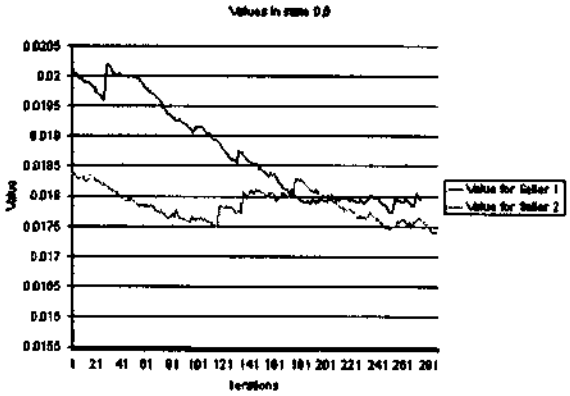

Figure 2: V(0,0) with only Informed Buyers

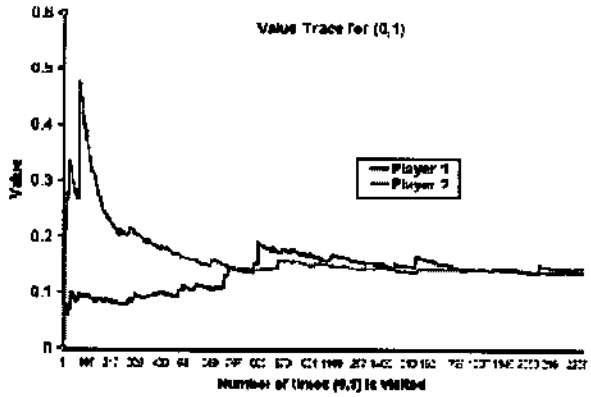

Figure 3: Plot of V(0,1)

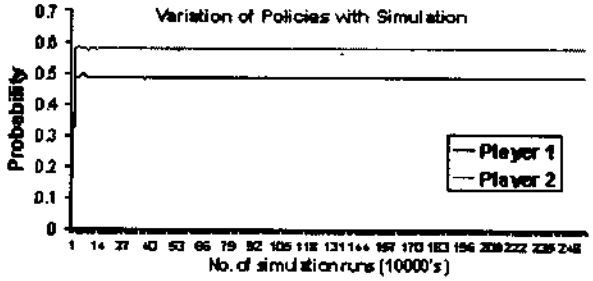

Figure 4: Policy Plot in the Infinite Capacity Game

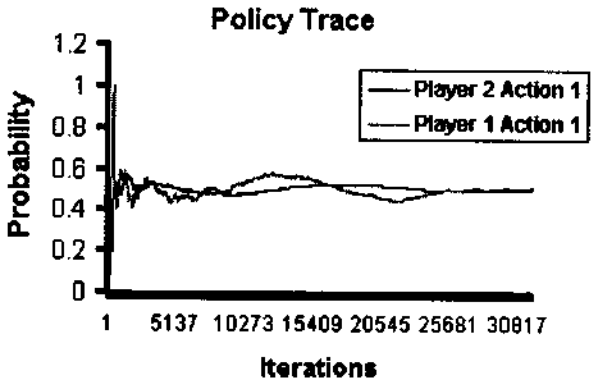

Figure 5: Convergence in the Two- Action Game

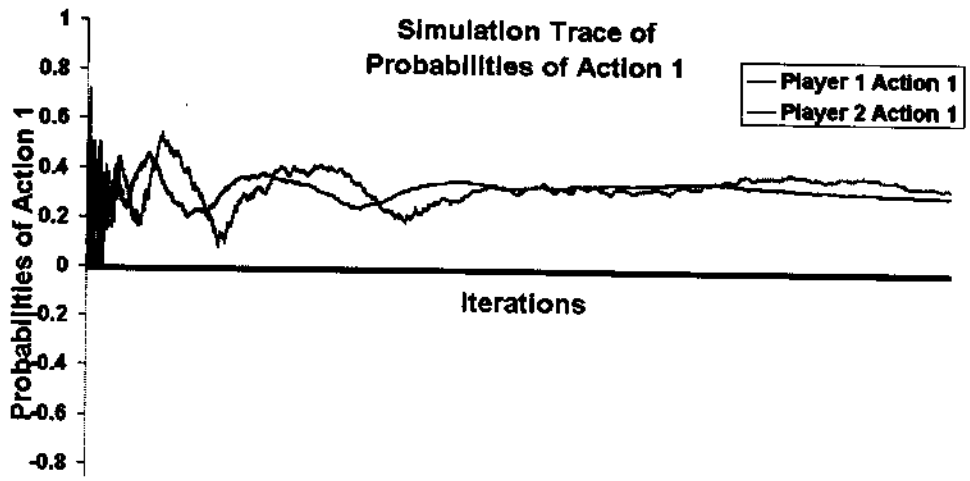

Figure 6: Convergence in the No-Balking Case