Autonomous Agents and Multi-Agent Systems, 5, 289–304, 2002
© 2002 Kluwer Academic Publishers. Manufactured in The Netherlands.

Pricing in Agent Economies Using Multi-Agent Q-Learning

GERALD TESAURO AND Tesauro@watson.ibm.com
JEFFREY O. KEPHART kephart@us.ibm.com
IBM Institute for Advanced Commerce, IBM Thomas J. Watson Research Center, Yorktown Heights, NY 10598, USA

Abstract. This paper investigates how adaptive software agents may utilize reinforcement learning algorithms such as Q-learning to make economic decisions such as setting prices in a competitive marketplace. For a single adaptive agent facing fixed-strategy opponents, ordinary Q-learning is guaranteed to find the optimal policy. However, for a population of agents each trying to adapt in the presence of other adaptive agents, the problem becomes non-stationary and history dependent, and it is not known whether any global convergence will be obtained, and if so, whether such solutions will be optimal. In this paper, we study simultaneous Q-learning by two competing seller agents in three moderately realistic economic models. This is the simplest case in which interesting multi-agent phenomena can occur, and the state space is small enough so that lookup tables can be used to represent the Q-functions. We find that, despite the lack of theoretical guarantees, simultaneous convergence to self-consistent optimal solutions is obtained in each model, at least for small values of the discount parameter. In some cases, exact or approximate convergence is also found even at large discount parameters. We show how the Q-derived policies increase profitability and damp out or eliminate cyclic price "wars" compared to simpler policies based on zero lookahead or short-term lookahead. In one of the models (the "Shopbot" model) where the sellers' profit functions are symmetric, we find that Q-learning can produce either symmetric or broken-symmetry policies, depending on the discount parameter and on initial conditions.

Keywords: machine learning, reinforcement learning, adaptive multi-agent systems, agent economies.

1. Introduction

Reinforcement Learning (RL) procedures have been established as powerful and practical methods for solving Markov Decision Problems. One of the most significant and actively investigated RL algorithms is Q-learning [15]. Q-learning is an algorithm for learning to estimate the long-term expected reward for a given state-action pair. It has the nice property that it does not need a model of the environment, and it can be used for on-line learning. Strong convergence of Q-learning to the exact optimal value functions and policies has been proven when lookup table representations of the Q-function are used [16]; this is feasible in small state spaces. In large state spaces where lookup tables are infeasible, RL methods can be combined with function approximators to give good practical performance despite the lack of theoretical guarantees of convergence to optimal policies.

Most real-world problems are not fully Markov in nature—they are often non-stationary, history-dependent and/or not fully observable. In order for RL methods to be more generally useful in solving such problems, they need to be extended to handle these non-Markovian properties. One important application domain where

the non-Markovian aspects are paramount is the area of multi-agent systems. This area is expected to be increasingly important in the future, due to the potential rapid emergence of "agent economies" consisting of large populations of interacting software agents engaged in various forms of economic activity. The problem of multiple agents simultaneously adapting is in general non-Markov, because each agent provides an effectively non-stationary environment for the other agents. Hence the existing convergence guarantees do not hold, and in general, it is not known whether any global convergence will be obtained, and if so, whether such solutions are optimal.

Some progress has been made in analyzing certain special case multiagent problems. For example, the problem of "teams," where all agents share a common objective function, has been studied, for example, in [1]. Likewise, the purely competitive case of zero-sum objective functions has been studied in [7], where an algorithm called "minimax-Q" was proposed for two-player zero-sum games, and shown to converge to the optimal value function and policies for both players. Sandholm and Crites studied simultaneous Q-learning by two players in the Iterated Prisoner's Dilemma game [9], and found that the learning procedure generally converged to stationary solutions. However, the extent to which those solutions were "optimal" was unclear. Recently, Hu and Wellman proposed an algorithm for calculating optimal Q-functions in two-player arbitrary-sum games [4]. This algorithm is an important first step. However, it does not yet appear to be useable for practical problems, because it assumes that policies followed by both players will be Nash equilibrium policies, and it does not address the "equilibrium coordination" problem, i.e., if there are multiple Nash equilibria, how do the agents decide which equilibrium to choose? We suspect that this may be a serious problem, since according to the "folk theorem of iterated games" [6], there can be a proliferation of Nash equilibria when there is sufficiently high emphasis on future rewards, i.e., a large value of the discount parameter γ. Furthermore, there may be inconsistencies between the assumed Nash policies, and the policies implied by the Q-functions calculated by the algorithm.

In this paper, we study simultaneous Q-learning in an economically motivated two-player game. The players are assumed to be two sellers of similar or identical products, who compete against each other on the basis of price. At each time step, the sellers alternately take turns setting prices, taking into account the other seller's current price. After the price has been set, the consumers then respond instantaneously and deterministically, choosing either seller 1's product or seller 2's product (or no product), based on the current price pair (p_1, p_2), leading to an instantaneous reward or profit (R_1, R_2) given to sellers 1 and 2 respectively. We assume initially that the sellers have full knowledge of the expected consumer demand for any given price pair, and in fact have full knowledge of both profit functions.

While it is true that a game-theoretic solution could be computed in this simplified full-information game, we expect that this will not be the case in real-world economies, due to lack of knowledge of buyer demand and of opponent profits and costs. Also, even if it were possible to compute a game-theoretic solution, it might not be valid since there is no assurance of full rationality of all opponents. Since machine learning methods can potentially address each of these limitations,

this provides impetus for investigating learning-based methods such as Q-learning. We stress that, in general, the Markov assumption needed to prove convergence of normal Q-learning will be violated in these economies. (A Q-learning agent would face an MDP only if the buyers and all opponent sellers utilized stationary policies.) However, this doesn't imply that Q-learning is expected to behave badly; it simply means that theory is not able to say how well Q-learning is expected to work. In the absence of theoretical guidance, we are attempting to develop an empirical understanding of multi-agent Q-learning in this study.

Our work builds on prior research reported in [12, 13]. Those papers examined the effect of including foresight, i.e. an ability to anticipate longer-term consequences of an agent's current action. Two different algorithms for agent foresight were presented: (i) a generalization of the minimax search procedure in two-player zero-sum games; (ii) a generalization of the Policy Iteration method from dynamic programming, in which both players' policies are simultaneously improved, until self-consistent policy pairs are obtained that optimize expected reward over two time steps. It was found that including foresight in the agents' pricing algorithms generally improved overall agent profitability, and usually damped out or eliminated the pathological behavior of unending cyclic "price wars," in which long episodes of repeated undercutting amongst the sellers alternate with large jumps in price. Such price wars were found to be rampant in prior studies of agent economy models [5, 8] when the agents use "myopically optimal" or "myoptimal" pricing algorithms that optimize immediate reward, but do not anticipate the longer-term consequences of an agent's current price setting.

Our motivation for studying simultaneous Q-learning in this paper is threefold. First, if Q-functions can be learned simultaneously and self-consistently for both players, the policies implied by those Q-functions should be self-consistently optimal. In other words, an agent will be able to correctly anticipate the longer-term consequences of its own actions, the other agents' actions, and will correctly model the other agents as having an equivalent capability. Hence the classic problem of infinite recursion of opponent models will be avoided. In contrast, in other approaches to adaptive multi-agent system, these issues are more problematic. For example, [14] proposed a recursive opponent modeling scheme, in which level-0 agents do no opponent modeling, level-1 agents model the opponents as being level-0, level-2 agents model the opponents as being level-1, etc. In this approach, there is no effective way for an agent to model other agents as being at an equivalent level of depth or complexity.

The second advantage of Q-learning is that the solutions should correspond to deep lookahead: in principle, the Q-function represents the expected reward looking infintely far ahead in time, exponentially weighted by a discount parameter $0 < \gamma < 1$. In contrast, the prior work of [13] was based on shallow finite lookahead. Finally, in comparison to directly modeling agent policies, the Q-function approach seems more extensible to the situation of very large economies with many competing sellers. Our intuition is that approximating Q-functions with nonlinear function approximators such as neural networks is more feasible than approximating the corresponding policies. Furthermore, in the Q-function approach, each agent only needs to maintain a single Q-function for itself, whereas in the policy modeling

approach, each agent needs to maintain a policy model for every other agent; the latter seems infeasible when the number of sellers is large.

The remainder of this paper is organized as follows. Section 2 describes the structure and dynamics of our model two-seller economy, and presents three economically-based models of seller profit (Price-Quality, Information-Filtering, and Shopbot) which are known to be prone to price wars when agents myopically optimize their short-term payoffs. We deliberately choose parameters to place each of these systems in a price-war regime. In Section 3, we describe details of how we implement Q-learning in these model economies. As a first step, we examine the simple case of ordinary Q-learning, where one of the two sellers uses Q-learning and the other seller uses a fixed pricing policy (the myopically optimal, or "myoptimal" policy). We then examine, in Section 4, the more interesting and novel situation of simultaneous Q-learning by both sellers. Finally, Section 5 summarizes the main conclusions and discusses promising directions and challenges for future work.

2. Model agent economies

Real agent economies are likely to contain large numbers of agents, with complex details of how the agents behave and interact with each other on multiple time scales. Our approach toward modeling and understanding such complexity is to begin by making a number of simplifying assumptions. We first consider the simplest possible case of two competing seller agents offering similar or identical products to a large population of consumer agents. The sellers compete on the basis of price, and we assume that prices are discretized and can lie between a minimum and maximum price, such that the number of possible prices is at most a few hundred. This renders the state space small enough that it is feasible to use lookup tables to represent the agents' pricing policies and expected profits. Time in the simulation is also discretized; at each time step, we assume that the consumers compare the current prices of the two sellers, and instantaneously and deterministically choose to purchase from at most one seller. Hence at each time step, for each possible pair of seller prices, there is a deterministic reward or profit given to each seller. The simulation can iterate forever, and there may or may not be a discounting factor for the present value of future rewards.

It is worth noting that the consumers are not regarded as "players" in the model. The consumers have no strategic role: they behave according to an extremely simple, fixed, short-term greedy rule (buy the lowest priced product at each time step), and are regarded as merely providing a stationary environment in which the two sellers can compete in a two-player game. This is clearly a simplifying first step in the study of multi-agent phenomena, and in future work, the models will be extended to include strategic and adaptive behavior on the part of the consumers as well. This will change the notion of "desirable" system behavior. In the present model, desirable behavior would resemble "collusion" between the two sellers in charging very high prices, so that both could obtain high profits. Obviously this is not desirable from the consumers' viewpoint.

Regarding the dynamics of seller price adjustments, we assume that the sellers alternately take turns adjusting their prices, rather than simultaneously setting prices (i.e., the game is extensive-form rather than normal-form). Our choice of alternating-turn dynamics is motivated by two considerations: (a) As the number of sellers becomes large and the model becomes more realistic, it seems more reasonable to assume that the sellers will adjust their prices at different times rather than at the same time, although they probably will not take turns in a well-defined order. (b) With alternating-turn dynamics, we can stay within the normal Q-learning framework where the Q-function implies a deterministic optimal policy: it is known that in two-player alternating turn games, there always exists a deterministic policy that is as good as any non-deterministic policy [7]. In contrast, in games with simultaneous moves (for example, rock-paper-scissors), it is possible that no deterministic policy is optimal, and that the existing Q-learning formalism for MDPs would have to be modified and extended so that it could yield non-deterministic optimal policies.

We study Q-learning in three different economic models that have been described in detail elsewhere [3, 5, 8]. The first model, called the "Price-Quality" model (Sairamesh and Kephart, 1998), models the sellers' products as being distinguished by different values of a scalar "quality" parameter, with higher-quality products being perceived as more valuable by the consumers. The consumers are modeled as trying to obtain the lowest-priced product at each time step, subject to threshold-type constraints on both quality and price, i.e., each consumer has a maximum allowable price and a minimum allowable quality. The similarity and substitutability of seller products leads to a potential for direct price competition; however, the "vertical" differentiation due to differing quality values leads to an asymmetry in the sellers' profit functions. It is believed that this asymmetry is responsible for the unending cyclic price wars that emerge when the sellers employ myoptimal pricing strategies.

The second model is an "Information-Filtering" model described in detail in [5]. In this model there are two competing sellers of news articles in somewhat overlapping categories. In contrast to the vertical differentiation of the Price-Quality model, this model contains a horizontal differentiation in the differing article categories. To the extent that the categories overlap, there can be direct price competition, and to the extent that they differ, there are asymmetries introduced that again lead to the potential for cyclic price wars.

The third model is the so-called "Shopbot" model described in [3], which is intended to model the situation on the Internet in which some consumers may use a Shopbot to compare prices of all sellers offering a given product, and select the seller with the lowest price. In this model, the sellers' products are exactly identical and the profit functions are symmetric. Myoptimal pricing leads the sellers to undercut each other until the minimum price point is reached. At that point, a new price war cycle can be launched, due to buyer asymmetries rather than seller asymmetries. The fact that not all buyers use the Shopbot, and some buyers instead choose a seller at random, means that it can be profitable for a seller to abandon the low-price competition for the bargain hunters, and instead maximally exploit the random buyers by charging the maximum possible price.

An example profit function that we study, taken from the Price-Quality model, is as follows: Let p_1 and p_2 represent the prices charged by seller 1 and seller 2 respectively. Let q_1 and q_2 represent their respective quality parameters, with $q_1 > q_2$. Let $c(q)$ represent the cost to a seller of producing an item of quality q. Then the profits to sellers 1 and 2 per item sold are given respectively by $p_1 - c(q_1)$ and $p_2 - c(q_2)$. The number of consumers that purchase from each seller is derived from the following model of buyer behavior: Each consumer i is assumed to have a quality threshold q_i s.t. all products with qualities greater than or equal to q_i are judged equally valuable, while qualities below q_i are judged unacceptable and the consumer refuses to buy them. The values of q_i are distributed throughout the buyer population according to a uniform random distribution between 0 and 1. There is also an additional global constraint on all consumers that they will refuse to buy a product with price greater than its quality value. Given these assumptions, [8] showed analytically that in the limit of infinitely many consumers, the instantaneous profits per consumer R_1 and R_2 obtained by seller 1 and seller 2 respectively are given by:

$$R_1 = \begin{cases} (q_1 - p_1)(p_1 - c(q_1)) & \text{if } 0 \le p_1 \le p_2 \text{ or } p_1 > q_2 \\ (q_1 - q_2)(p_1 - c(q_1)) & \text{if } p_2 < p_1 < q_2 \end{cases} \quad [1]$$

$$R_2 = \begin{cases} (q_2 - p_2)(p_2 - c(q_2)) & \text{if } 0 \le p_2 < p_1 \\ 0 & \text{if } p_2 \ge p_1 \end{cases} \quad [2]$$

A plot of the profit landscape for seller 1 as a function of prices p_1 and p_2 is given in figure 1, for the following parameter settings: $q_1 = 1.0$, $q_2 = 0.9$, and $c(q) = 0.1(1 + q)$. (These specific parameter settings were chosen because they are known to generate harmful price wars when the agents use myopic optimal pricing.) We can see in this figure that the myopic optimal price for seller 1 as a function of seller 2's price, $p_1^*(p_2)$, is obtained for each value of p_2 by sweeping across all values of p_1 and choosing the value that gives the highest profit. We can see that for small values of p_2, the peak profit is obtained at $p_1 = 0.9$, whereas for larger values of p_2, there is eventually a discontinuous shift to the other peak, which follows along the parabolic-shaped ridge in the landscape. An analytic expression for the myopic optimal price for seller 1 as a function of p_2 is as follows (defining $x_1 = q_1 + c(q_1)$ and $x_2 = q_2 + c(q_2)$:

$$p_1^*(p_2) = \begin{cases} q_2 & \text{if } 0 \le p_2 < x_1 - q_2 \\ p_2 & \text{if } x_1 - q_2 \le p_2 \le \frac{1}{2}x_1 \\ \frac{1}{2}x_1 & \text{if } p_2 > \frac{1}{2}x_1 \end{cases} \quad [3]$$

Similarly, the myopic optimal price for seller 2 as a function of the price set by seller 1, $p_2^*(p_1)$, is given by the following formula (assuming that prices are discrete

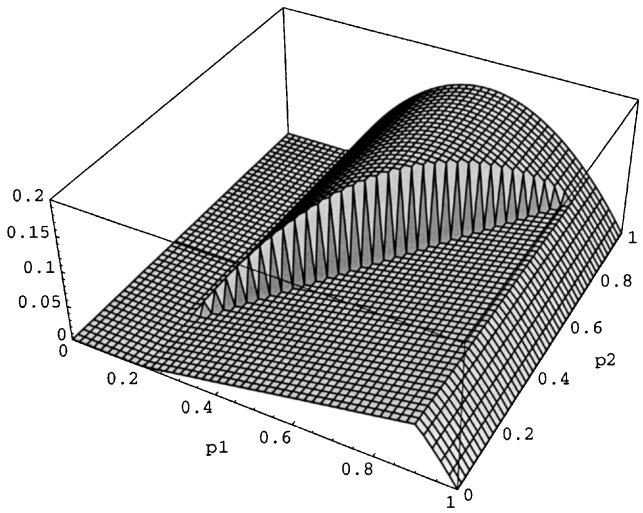

Figure 1. Sample profit landscape for seller 1 in Price-Quality model, as a function of seller 1 price p_1 and seller 2 price p_2.

and that ϵ is the price discretization interval):

$$p_2^*(p_1) = \begin{cases} c(q_2) & 0 \le p_1 \le c(q_2) \\ p_1 - \epsilon & \text{if } c(q_2) \le p_1 \le \frac{1}{2}x_2 \\ \frac{1}{2}x_2 & \text{if } p_1 > \frac{1}{2}x_2 \end{cases} \quad [4]$$

We also note in passing that there are similar profit landscapes for each of the sellers in the Information-Filtering model and in the Shopbot model. In all three models, it is the existence of multiple, disconnected peaks in the landscapes, with relative heights that can change depending on the other seller's price, that leads to price wars when the sellers behave myopically.

Regarding the information set that is made available to the sellers, we have made a simplifying assumption as a first step that the players have essentially perfect information. They can model the consumer behavior perfectly, and they also have perfect knowledge of each other's costs and profit functions. Hence our model is thus a two-player perfect-information deterministic game that is very similar to games like chess. The main differences are that the profits in our model are not strictly zero-sum, and that there are no terminating or absorbing nodes in our model's state space. Also in our model, payoffs are given to the players at every time step, whereas in games such as chess, payoffs are only given at the terminating nodes.

As mentioned previously, we constrain the prices set by the two sellers to lie in a range from some minimum to maximum allowable price. The prices are discretized, so that one can create lookup tables for the seller profit functions $U(p_1, p_2)$. Furthermore, the optimal pricing policies for each seller as a function of the other seller's price, $p_1^*(p_2)$ and $p_2^*(p_1)$, can also be represented in the form of table lookups.

3. Single-agent Q-learning

We first consider ordinary single-agent Q-learning in the above two-seller economic models. The procedure for Q-learning is as follows. Let $Q(s, a)$ represent the discounted long-term expected reward to an agent for taking action a in state s. The discounting of future rewards is accomplished by a discount parameter γ such that the value of a reward expected at n time steps in the future is discounted by γ^n. Assume that the $Q(s, a)$ function is represented by a lookup table containing a value for every possible state-action pair, and assume that the table entries are initialized to arbitrary values. Then the procedure for solving for $Q(s, a)$ is to infinitely repeat the following two-step loop:

1. Select a particular state s and a particular action a, observe the immediate reward r for this state-action pair, and observe the resulting state s'.
2. Adjust $Q(s, a)$ according to the following equation:

$$\Delta Q(s, a) = \alpha\Big[r + \gamma \max_b Q(s', b) - Q(s, a)\Big] \qquad [5]$$

where α is the learning rate parameter, and the max operation represents choosing the optimal action b among all possible actions that can be taken in the successor state s' leading to the greatest Q-value. A wide variety of methods may be used to select state-action pairs in step 1, provided that every state-action pair is visited infinitely often. For any stationary Markov Decision Problem, the Q-learning procedure is guaranteed to converge to the correct values, provided that α is decreased over time with an appropriate schedule.

We first consider using Q-learning for one of the two sellers in our economic models, while the other seller maintains a fixed pricing policy. In the simulations described below the fixed policy is in fact the my-optimal policy p^* represented for example in the Price-Quality model by Eqs. [3] and [4].

In our pricing application, the distinction between states and actions is somewhat blurred. We will assume that the "state" for each seller is sufficiently described by the other seller's last price, and that the "action" is the current price decision. This should be a sufficient state description because no other history is needed either for the determination of immediate reward, or for the calculation of the myoptimal price by the fixed-strategy player. We have also modified the concepts of immediate reward r and next-state s' for the two-agent case. We define s' as the state that is obtained, starting from s, of one action by the Q-learner and a response action by the fixed-strategy opponent. Likewise, the immediate reward is defined as the sum of the two rewards obtained after those two actions. These modifications were introduced so that the state s' would have the same player to move as state s. (A possible alternative to this, which we have not investigated, is to include the side-to-move as additional information in the state-space description.)

In the simulations reported below, the sequence of state-action pairs selected for the Q-table updates were generated by uniform random selection from amongst all possible table entries. The initial values of the Q-tables were generally set to the

immediate reward values. (Consequently the initial Q-derived policies corresponded to myoptimal policies.) The learning rate was varied with time according to:

$$\alpha(t) = \alpha(0)/(1 + \beta t) \quad [6]$$

where the initial learning rate $\alpha(0)$ was usually set to 0.1, and the constant $\beta \sim 0.01$ when the simulation time t was measured in units of N^2, the size of the Q-table. (N is the number of possible prices that could be selected by either player.) A number of different values of the discount parameter γ were studied, ranging from $\gamma = 0$ to $\gamma = 0.9$.

Results for single-agent Q-learning in all three models indicated that Q-learning worked well (as expected) in each case. In each model, for each value of the discount parameter, exact convergence of the Q-table to a stationary optimal solution was found. The convergence times ranged from a few hundred sweeps through each table element, for smaller values of γ, to at most a few thousand updates for the largest values of γ. In addition, once Q-learning converged, we then measured the expected cumulative profit of the policy derived from the Q-function. We ran the Q-policy against the other player's myopic policy from 100 random starting states, each for 200 time steps, and averaged the resulting cumulative profit for each player. We found that, in each case, the seller achieved greater profit against a myopic opponent by using a Q-derived policy than by using a myopic policy. (This was true even for $\gamma = 0$, because, due to the redefinition of Q updates summing over two time steps, the case $\gamma = 0$ effectively corresponds to a two-step optimization, rather than the one-step optimization of the myopic policies.) Furthermore, the cumulative profit obtained with the Q-derived policy monotonically increased with the increasing γ (as expected).

It was also interesting to note that in many cases, the expected profit of the myopic opponent also increased when playing against the Q-learner, and also improved monotonically with increasing γ. The explanation is that, rather than better exploiting the myopic opponent, as would be expected in a zero-sum game, the Q-learner instead reduced the region over which it would participate in a mutually undercutting price war. Typically we find in these models that with myopic vs. myopic play, large-amplitude price wars are generated that start at very high prices and persist all the way down to very low prices. When a Q-learner competes against a myopic opponent, there are still price wars starting at high prices, however, the Q-learner abandons the price war more quickly as the prices decrease. The effect is that the price-war regime is smaller and confined to higher average prices, leading to a closer approximation to collusive behavior, with greater expected utilites for both players.

An illustrative example of the results of single-agent Q-learning is shown in Figure 2. Figure 2(a) plots the average profit for both sellers in the Shopbot model, when one of the sellers is myopic and the other is a Q-learner. (As the model is symmetric, it doesn't matter which seller is the Q-learner.) Figure 2(b) plots the myopic price curve of seller 2 against the Q-derived price curve (at $\gamma = 0.5$) of seller 1. We can see that both curves have a maximum price of 1 and a minimum price of approximately 0.58. The portion of both curves lying along the diagonal

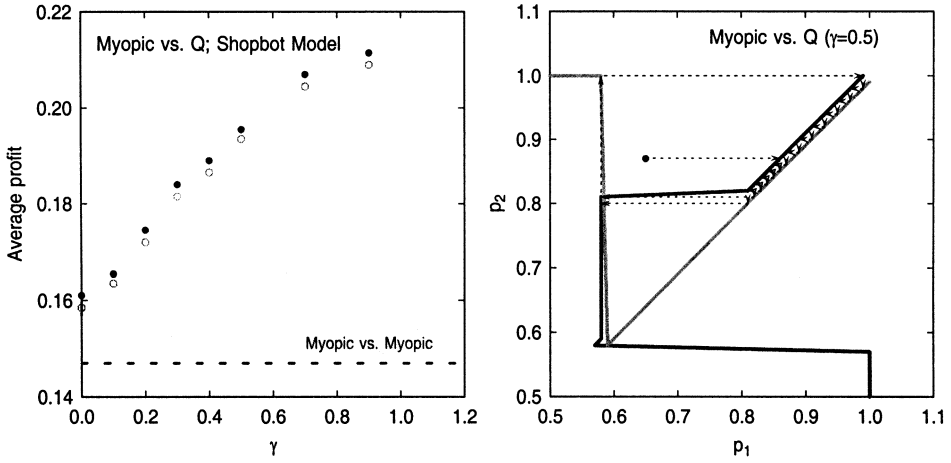

Figure 2. Results of single-agent Q-learning in the Shopbot model. (a) Average profit per time step for Q-learner (seller 1, filled circles) and myopic seller (seller 2, open circles) vs. discount parameter γ. Dashed line indicates baseline expected profit when both sellers are myopic. (b) Cross-plot of Q-derived price curve (seller 1) vs. myopic price curve (seller 2) at $\gamma = 0.5$. Dashed line and arrows indicate a temporal price-pair trajectory using these policies, starting from filled circle.

indicates undercutting behavior, in which case the seller will respond to the opponent's price by undercutting by ϵ, the price discretization interval.

The system dynamics for the state (p_1, p_2) in Figure 2(b) can be obtained by alternately applying the two pricing policies. This can be done by a simple iterative graphical construction, in which for any given starting point, one first holds p_2 constant and moves horizontally to the $p_1(p_2)$ curve, and then one holds p_1 constant and moves vertically to the $p_2(p_1)$ curve. We see in this figure that the iterative graphical construction leads to an unending cyclic price war, whose trajectory is indicated by the dashed line. Note that the price-war behavior begins at the price pair (1, 1), and persists until a price of approximately 0.83. At this point, seller 1 abandons the price war, and resets its price to 1, leading once again to another round of undercutting.

The amplitude of this price war is diminished compared to the situation in which both players use a myopic policy. In that case, seller 1's curve would be a mirror image of seller 2's curve, and the price war would persist all the way to the minimum price point, leading to a lower expected profit for both sellers.

4. Multi-agent Q-learning

We now examine the more interesting and challenging case of simultaneous training of Q-functions and policies for both sellers. Our approach is to use the same formalism presented in the previous section, and to alternately adjust a random entry in seller 1's Q-function, followed by a random entry in seller 2's Q-function. As each seller's Q-function evolves, the seller's pricing policy is correspondingly updated so that it optimizes the agent's current Q-function. In modeling the two-step payoff r

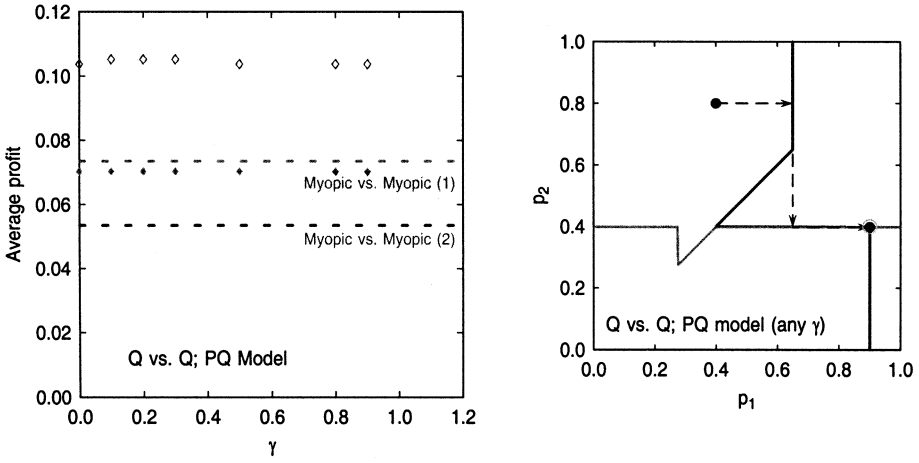

Figure 3. Results of simultaneous Q-learning in the Price-Quality model. (a) Average profit per time step for seller 1 (solid diamonds) and seller 2 (open diamonds) vs. discount parameter γ. Dashed line indicates baseline myopic vs. myopic expected profit. Note that seller 2's profit is higher than seller 1's, even though seller 2 has a lower quality parameter. (b) Cross-plot of Q-derived price curves (at any γ). Dashed line and arrows indicate a sample price dynamics trajectory, starting from the filled circle. The price war is eliminated and the dynamics evolves to a fixed point indicated by an open circle.

to a seller in Eq. [5], we use the opponent's current policy as implied by its current Q-function. In experimenting with different learning parameter values, it was found that the same parameter values used in the previous section also gave good results in the multi-agent case. In most of the experiments, the Q-functions were initialized to the instantaneous payoff values (so that the policies corresponded to myopic policies), although other initial conditions were explored in a few experiments.

For simultaneous Q-learning in the Price-Quality model, we find robust convergence to a unique pair of pricing policies, independent of the value of γ, as illustrated in Figure 3(b). This solution also corresponds to the solution found by generalized minimax and by generalized DP in [13]. We note that repeated application of this pair of price curves leads to a dynamical trajectory that eventually converges to a fixed-point located at ($p_1 = 0.9$, $p_2 = 0.4$). A detailed analysis of these pricing policies and the fixed-point solution is presented in (Tesauro and Kephart, 1999). In brief, for sufficiently low prices of seller 2, it pays seller 1 to abandon the price war and to charge a very high price, $p_1 = 0.9$. The value of $p_2 = 0.4$ then corresponds to the highest price that seller 2 can charge without provoking an undercut by seller 1, based on a two-step lookahead calculation (seller 1 undercuts, and then seller 2 replies with a further undercut). We note that this fixed point does not correspond to a Nash equilibrium, since both players have an incentive to deviate, based on a one-step lookahead calculation. In fact, one can show that there is no pure-strategy Nash equilibrium in this particular game, and that this is the reason why myopic vs. myopic play leads to price-war cycles. (Otherwise, the myopic play would lead to a fixed point at the Nash equilibrium.) Instead, it was conjectured in [13] that the solution observed in Figure 3(b) corresponds to a "subgame-perfect" equilibrium [2] rather than a Nash equilibrium. The notion of

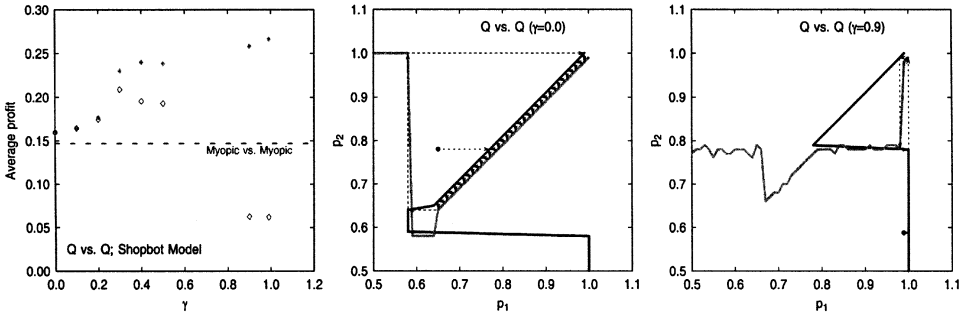

Figure 4. Results of simultaneous Q-learning in the Shopbot model. (a) Average profit per time step for seller 1 (solid diamonds) and seller 2 (open diamonds) vs. discount parameter γ. Dashed line indicates baseline myopic vs. myopic expected profit. (b) Cross-plot of Q-derived price curves at $\gamma = 0$; the solution is symmetric. Dashed line and arrows indicate a sample price dynamics trajectory. (c) Cross-plot of Q-derived price curves at $\gamma = 0.9$; the solution is asymmetric.

subgame-perfect equilibria applies to extensive-form games where the players alternately take turns. It is normally thought of as the strategy obtained by backwards reasoning from the terminal states of a game to deduce the optimal play at n steps from the end, assuming subgame-perfect play thereafter in the remaining $(n-1)$ steps. The important distinction here is that the game is of infinite length and there are no terminating states; this is why the conjecture of [13] was not proven.

The cumulative profits obtained by the pair of pricing policies are plotted in Figure 3(a). It is interesting that seller 2, the lower-quality seller, actually obtains a significantly higher profit than seller 1, the higher-quality seller. In contrast, with myopic vs. myopic pricing, seller 2 does worse than seller 1.

In the Shopbot model, we did not find exact convergence of the Q-functions for each value of γ. However, in those cases where exact convergence was not found, we did find very good approximate convergence, in which the Q-functions and policies converged to stationary solutions to within small random fluctuations. Different solutions were obtained at each value of γ. We generally find that a symmetric solution, in which the shapes of $p_1(p_2)$ and $p_2(p_1)$ are identical, is obtained at small γ, whereas a broken symmetry solution, similar to the Price-Quality solution, is obtained at large γ. We also found a range of γ values, between 0.1 and 0.2, where either a symmetric or asymmetric solution could be obtained, depending on initial conditions. The asymmetric solution was counter-intuitive to us, because we expected that the symmetry of the two sellers' profit functions would lead to a symmetric solution. In hindsight, we can apply the same type of reasoning as in the Price-Quality model to explain the asymmetric solution. A plot of the expected profit for both sellers as a function of γ is shown in Figure 4(a). Plots of the symmetric and asymmetric solution, obtained at $\gamma = 0$ and $\gamma = 0.9$ respectively, are shown in Figures 4(b) and 4(c).

Finally, in the Information-Filtering model, we found that simultaneous Q-learning produced exact or good approximate convergence for small values of γ ($0 \leq \gamma \leq 0.5$). For large values of γ, no convergence was obtained. The simultaneous Q-learning solutions yielded reduced-amplitude price wars, and

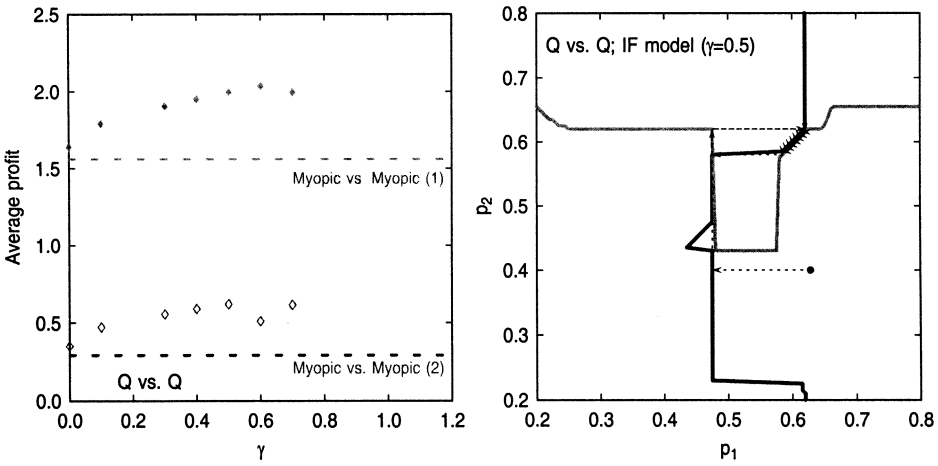

Figure 5. Results of multi-agent Q-learning in the Information-Filtering model. (a) Average profit per time step for seller 1 (solid diamonds) and seller 2 (open diamonds) vs. discount parameter γ. (The data points at $\gamma = 0.6, 0.7$ represent unconverged Q-functions and policies.) Dashed lines indicates baseline expected profit when both sellers are myopic. (b) Cross-plot of Q-derived price curves at $\gamma = 0.5$.

montonically increasing profitability for both sellers as a function of γ, at least up to $\gamma = 0.5$. A few data points were examined at $\gamma > 0.5$, and even though there was no convergence, the Q-policies still yielded greater profit for both sellers than in the myopic vs. myopic case. A plot of the Q-derived policies and system dynamics for $\gamma = 0.5$ is shown in figure 5(b). The expected profits for both players as a function of γ is plotted in Figure 5(a).

5. Conclusions

We have examined single-agent and multi-agent Q-learning in three models of a two-seller economy in which the sellers alternately take turns setting prices, and then instantaneous profits are given to both sellers based on the current price pair. Such models fall into the category of two-player, alternating-turn, arbitrary-sum Markov games, in which both the rewards and the state-space transitions are deterministic. The game is Markov because the state space is fully observable and the rewards are not history dependent.

In all three models (Price-Quality, Information-Filtering, and Shopbot), large-amplitude cyclic price wars are obtained when the sellers myopically optimize their instantaneous profits without regard to longer-term impact of their pricing policies. We find that, in all three models, the use of Q-learning by one of the sellers against a myopic opponent invariably results in exact convergence to the optimal Q-function and optimal policy against that opponent, for all allowed values of the discount parameter γ. The use of the Q-derived policy yields greater expected profit for the Q-learner, with monotonically increasing profit as γ increases. In many cases, it has a side benefit of also enhancing the welfare of the myopic opponent. This comes

about by reducing the amplitude of the undercutting price-war regime, or in some cases, eliminating it completely.

We have also studied the more interesting and challenging situation of simultaneously training Q-functions for both sellers. This is more difficult because as each seller's Q-function and policy change, it provides a non-stationary environment for adaptation of the other seller. No convergence proofs exist for such simultaneous Q-learning by multiple agents. Nevertheless, despite the absence of theoretical guarantees, we do find generally good behavior of the algorithm in our model economies. In two of the models (Shopbot and Price-Quality), we find exact or very good approximate convergence to simultaneously self-consistent Q-functions and optimal policies for any value of γ, whereas in the Information-Filtering model, simultaneous convergence was found for $\gamma \leq 0.5$. In the Information-Filtering and Shopbot models, monotonically increasing expected profits for both sellers were also found for small values of γ. In the Price-Quality model, simultaneous Q-learning yields an asymmetric solution, corresponding to the solution found in [13], that is highly advantageous to the lesser-quality seller, but, slightly disadvantageous to the higher-quality seller, when compared to myopic vs. myopic pricing. A similar asymmetric solution is also found in the Shopbot model for large γ, even though the profit functions for both players are symmetric.

For each model, there exists a range of discount parameter values where the solutions obtained by simultaneous Q-learning are self-consistently optimal, and outperform the solutions obtained in [13]. This is presumably because the previously published methods were based on limited lookahead, whereas the Q-functions in principle look ahead infinitely far, with appropriate discounting.

It is intruiging that simultaneous Q-learning works well in our models, despite the lack of theoretical convergence proofs. Sandholm and Crites also found that simultaneous Q-learning generally converged in the Iterated Prisoner's Dilemma game. These empirical findings suggest that a deeper theoretical analysis of simultaneous Q-learning may be worth investigating. There may be some underlying theoretical principles that can explain why simultaneous Q-learning works, for at least certain classes of arbitrary-sum profit functions.

Several important challenges will also be faced in extending our approach to larger-scale, more realistic simulations. While there are some economic situations in the real world where there are only two dominant sellers, in general the number of sellers can be much greater. The situation that we foresee in agent economies is that the number of competing sellers will be very large. In this case, the seller profits and pricing functions will have such high input dimensionality that it will be infeasible to use lookup table state-space representations, and most likely some sort of compact representation combined with a function approximation scheme will be necessary. Preliminary results recently reported in [10] show that Q-learning combined with a simple regression-tree function approximation scheme can match the quality of the lookup table results in these models while providing significant advantages in storage space and in amount of training data required.

The utilization of any prior domain knowledge could also make the effective dimensionality much smaller and the learning task faced by the learning agent much easier. For example, using prior knowledge we know that learning a seller's

immediate reward function does not require a full n-dimensional price vector, but instead only depends on the seller's own price and on how many competitor prices are greater than, equal to, or less than the seller's price.

Furthermore, with many sellers, the concept of sellers taking turns adjusting their prices in a well-defined order becomes problematic. This could lead to an additional combinatorial explosion, if the mechanism for calculating expected reward has to anticipate all possible orderings of opponent responses.

While our economic models have a moderate degree of realism in their profit functions, they are unrealistic in the assumptions of knowledge and dynamics. In the work reported here, the state space was fully observable infinitely frequently at zero cost and with zero propagation delays. The expected consumer demand for a given price pair was instantaneous, deterministic and fully known to both players. Indeed, the players' exact profit functions were fully known to both players. It was also assumed that the players would alternately take turns equally often in a well-defined order in adjusting their prices. Under such assumptions of knowledge and dynamics, one could hope to develop an algorithm that could calculate in advance something like a game-theoretic optimal pricing algorithm for each agent.

However, in realistic agent economies, it is likely that agents will have much less than full knowledge of the state of the economy. Agents may not know the details of other agents' profit functions, and indeed an agent may not know its own profit function, to the extent that buyer behavior is unpredictable, The dynamics of buyers and sellers may also be more complex, random and unpredictable than what we have assumed here. There may also be information delays for both buyers and sellers, and part of the economic game may involve paying a cost in order to obtain information about the state of the economy faster and more frequently, and in greater detail. Finally, we expect that buyer behavior will be non-stationary, so that there will be a more complex co-evolution of buyer and seller strategies.

While such real-world complexities are daunting, there are reasons to believe that learning approaches such as Q-learning may play a role in practical solutions. The advantage of Q-learning is that one does not need a model of either the instantaneous payoffs or of the state-space transitions in the environment. One can simply observe actual rewards and transitions and base learning on that. While the theory of Q-learning requires exhaustive exploration of the state space to guarantee convergence, this may not be necessary when function approximators are used. In that case, after training a function approximator on a relatively small number of observed states, it may then generalize well enough on the unobserved states to give decent practical performance. Several recent empirical studies have provided evidence of this [1, 11, 17].

Acknowledgments

The authors thank Amy Greenwald for helpful discussions regarding the Shopbot model.

References

1. R. H. Crites and A. G. Barto, "Improving elevator performance using reinforcement learning," in D. Touretzky et al. (eds.), *Advances in Neural Information Processing Systems*, MIT Press, 1996, vol. 8, pp. 1017–1023.
2. D. Fudenberg and J. Tirole, *Game Theory*, MIT Press: Cambridge, MA: 1991.
3. A. Greenwald and J. O. Kephart, "Shopbots and pricebots," to appear in: *Proc. IJCAI-99*, 1999.
4. J. Hu and M. P. Wellman, "Multiagent reinforcement learning: theoretical framework and an algorithm," *Proc. ICML-98*, 1998.
5. J. O. Kephart, J. E. Hanson and, J. Sairamesh, "Price-war dynamics in a free-market economy of software agents," in *Proc. ALIFE-VI*, Los Angeles, 1998.
6. D. Kreps, *A Course in Microeconomic Theory*, Princeton Univ. Press: Princeton, NJ, 1990.
7. M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," *Proc. Eleventh Int. Conf. Machine Learning*, Morgan Kaufmann, 1994, pp. 157–163.
8. J. Sairamesh and J. O. Kephart, "Dynamics of price and quality differentiation in information and computational markets," *Proc. First Int. Conf. Information and Computation Economics (ICE-98)*, ACM Press, 1998, pp. 28–36.
9. T. W. Sandholm and R. H. Crites, "On multiagent Q-Learning in a semi-competitive domain," *14th Int. Joint Conf. Artificial Intelligence (IJCAI-95)* Workshop on Adaptation and Learning in Multiagent Systems, Montreal, Canada, 1995, pp. 71–77.
10. M. Sridharan and G. Tesauro, "Multi-agent Q-learning and regression trees for automated pricing decisions," *Proc. ICML-00*, to appear, 2000.
11. G. Tesauro, "Temporal difference learning and TD-Gammon," *Comm. of the ACM*, vol. 38, no. 3, pp. 58–67, 1995.
12. G. J. Tesauro and J. O. Kephart, "Foresight-based pricing algorithms in an economy of software agents," *Proc. First Int. Conf. Information and Computation Economics (ICE-98)*, ACM Press, 1998, pp. 37–44.
13. G. J. Tesauro and J. O. Kephart, "Foresight-based pricing algorithms in agent economies," *Decision Support Sciences*, to appear, 1999.
14. J. M. Vidal and E. H. Durfee, "Learning nested agent models in an information economy," *J. Experimental and Theoretical AI*, to appear, 1998.
15. C. J. C. H. Watkins, "Learning from delayed rewards," Ph.D. thesis, Cambridge University, 1989.
16. C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, pp. 279–292, 1992.
17. W. Zhang and T. G. Dietterich, "High-performance job-shop scheduling with a time-delay TD(λ) network." in D. Touretzky et al. (eds.), Advances in Neural Information Processing Systems, am Press, 1996, vol. 8, pp. 1024–1030.