

Foresight-based pricing algorithms in an economy of software agents

Gerald J. Tesauro and Jeffrey O. Kephart

IBM T. J. Watson Research Center

30 Saw Mill River Rd., Hawthorne NY, 10532

e-mail: tesauro@watson.ibm.com, kephart@watson.ibm.com

Abstract

We propose several heuristic approaches to the development of pricing algorithms for software agents that incorporate foresight, i.e., an ability to model and predict responses by competitors. In the absence of foresight, prior work has shown that, in an economy of myopic software agents, undesirable system behaviors such as endless price wars can frequently occur (Kephart et al., 1998). We show how the introduction of even the smallest amount of lookahead in the agents' pricing algorithms can significantly reduce or eliminate the occurrence of price wars. We also investigate two approaches to developing algorithms that are capable of deep lookahead, while avoiding the classic problem of infinite recursion of opponent models. The two approaches are based on adaptations of: (i) the classic minimax fixed-depth search algorithms used in two-player games such as chess; (ii) dynamic programming (DP)-style algorithms, that have recently been extended to the domain of two-player zero-sum Markov games (Littman, 1994).

1 Introduction

In prior work (Kephart et al., 1998; Sairamesh and Kephart, 1998) it has been shown that in large-scale economies of software agents, the potential exists for unending cycles of disastrous competitive "wars" in price/product space. Such price and niche wars are disastrous not only for the sellers in such economies (e.g. information sellers in an information filtering economy), but they can also be disastrous for the consumers as well. Price wars have the potential to be more rampant in agent economies than what we have observed in human economies due to a number of differences between agent and human economic players. Among such differences are: (1) greater ability of humans to predict long-term consequences of their price-setting actions; (2) reduced frictional effects such as consumer inertia in agent economies; and (3) reduced localization effects due to much greater connectivity offered by the Internet.

In this paper, we focus specifically on the use of foresight or predictive capability by software agents to anticipate re-

taliatory responses that will be made by other agents. Our intuition is that this is the most important factor in models studied so far that generate price wars. In (Kephart et al., 1998), it was shown that price-war phenomena are generally obtained in agent economies when the agents can do a good job of optimizing their immediate short-term reward or utility, but have no capability to predict system behavior beyond the current time step. Such "myopic optimal" agents may be tempted to engage in price-undercutting behavior to obtain an immediate short-term advantage. However, if the other agents retaliate by further undercutting, then the agent may be worse off in the long term than if it had kept its price at a high level and not attempted any undercutting. Our intuition is that if agents can be endowed with some sort of predictive capability that could anticipate longer-term consequences of current pricing behavior, then this could allow the agents to realize the futility of undercutting. Implementing foresight mechanisms on a large scale throughout an agent economy might then lead to radically different system behavior, and price wars might be greatly reduced or eliminated.

In considering algorithmic approaches to agent foresight, one set of issues that arises has to do with which type of algorithms will enable agents to make the most accurate predictions, given the constraints of a limited information set and limited computational resources. An additional set of issues has to do with what sort of collective system consequences result when a significant fraction of the agents base their actions on predictions. Will a society of machine learners converge to something like a game-theoretic solution (for example, a Nash equilibrium), or will they just endlessly chase one another's tails? The former question has been studied, for example, in (Milgrom and Roberts, 1991) and (Foster and Vohra, 1997), while the latter issues are beginning to be addressed, for example, in (Hu and Wellman, 1996) and (Vidal and Durfee, 1998).

In the present work, we are primarily concerned with issues of the depth and accuracy of agent lookahead. That is to say, how far ahead in time can an agent reliably predict those aspects of the future system behavior that are relevant to their current decisions, and how much lookahead is necessary to avoid pathological behavior? Is shallow lookahead sufficient, or is it necessary for agents to engage in deep lookahead in order to avoid price wars? Likewise, we would like to know how accurately an agent can predict, and how much accuracy is needed to avoid pathological behavior? It may be the case that only a coarse prediction is needed to avoid price wars. (For example, will any other agent set a price lower than mine on the next time step?)

Our proposed algorithms for agent foresight are designed to avoid the classic problem of infinite recursion of opponent models. That is to say, when modeling other agents, one needs to take into account the fact that those other agents are themselves using models of other agents, and that those models need to take into account that the other agents are using models, etc.. This can lead not only to logical problems in setting up the agent models, but also to greatly increasing levels of computational complexity with the depth of recursion. For example, in the work of (Vidal and Durfee, 1998), a recursive modeling scheme is proposed in which 0-level agents do no opponent modeling, 1-level agents model the other agents as being 0-level agents, 2-level agents model the other agents as being 1-level agents, etc.. In this scheme, the computational requirements greatly increase with the level of modeling, and furthermore, there is no adequate way for an agent to model other agents as being at the same level of depth.

We consider two basic heuristic approaches to avoiding an infinite recursion of opponent models. The first approach is adapted from the domain of two-player zero-sum games such as chess, in which full-width minimax search to a fixed finite depth has been found to be an effective algorithm. In this case, the infinite recursion is cut off by the finite depth of the search. Minimax search is only guaranteed to find optimal moves if the search goes all the way to the end of the game; however, it does seem to work well in practice for searches of lesser depth. For example, the chess machines Deep Thought and Deep Blue give the impression of generating sophisticated positional understanding as an emergent property from deep searches plus simple positional knowledge built into the evaluation function.

The second approach that we explore is to adapt algorithms from the fields of Dynamic Programming (DP) and Reinforcement Learning (RL); such algorithms have been found to work well for single agents in stationary Markov environments (i.e. Markov Decision Problems, or MDPs). The basic idea of DP/RL is "Policy Iteration" (Bertsekas, 1995), in which one starts with an initial policy, computes the value function induced by that policy, and then computes an improved policy that is greedy with respect to that value function. Policy iteration is guaranteed to converge to the optimal agent policy for single-agent MDPs. Recently there has been some work generalizing DP-type algorithms to two-player Markov games. For example, (Littman, 1994) introduced an algorithm called minimax-Q for two-player zero-sum games in which the players alternately take turns moving, which is guaranteed to converge to the optimal policies for both players. Unfortunately, we can't directly use minimax-Q in agent economies because the agent utilities are not strictly zero-sum. We investigate in this paper several heuristic DP-like approaches which can be used in arbitrary-sum games.

As a general caveat, we should point out that both classes of algorithms mentioned above seek *deterministic* optimal policies. (Here "optimal" is used in the game-theoretic sense of the best worst-case behavior against all possible opponent strategies.) However, it may be the case that such deterministic policies do not exist. For example, in the game of rock-paper-scissors, any deterministic policy can be defeated, and the best policies are non-deterministic and cannot be computed by these methods.

2 Summary of the utility landscape model

As a first step in the development of general foresight algorithms, we first consider the simplest possible case of two competing sellers, who alternately take turns adjusting their prices. We assume that the products offered by the two sellers are somewhat similar, leading to some potential for price competition between them, but there is also a degree of product differentiation, so that the cost and utility functions for the two sellers are in general asymmetric. There are several different economic models in which such asymmetries can come about. The primary model that we work with is a price-quality model that is described in detail in (Sairamesh and Kephart, 1998). In this model, products offered by different sellers are distinguished by different values of a "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 other model that we have studied is an information-filtering model described in detail in (Kephart et al., 1998). In this model there are two competing sellers of news articles in somewhat overlapping categories. The partial overlap of the categories leads to a potential for direct price competition; however, the fact that they are not identical introduces an element of product differentiation similar to the quality differentiation in the price-quality model. This product differentiation leads to an asymmetry in the optimal pricing strategies for the two sellers. At each time step, the consumers decide to subscribe to one of the two sellers, based on price and on the consumer's particular interest categories.

In both of the models mentioned above, the consumers deterministically and instantaneously choose one of the two sellers at every time step, based on the prices of the two sellers. This means that the only relevant variables in the state space description are the prices of the two sellers at each time step. The two sellers alternately take turns adjusting their prices at each time step, and then depending on the particular prices set, the resulting consumer behavior determines the amount of "profit" or "utility" obtained by each seller. The simulation can iterate forever, and there may or may not be a discounting factor for the present value of future rewards.

An example utility 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 assuming the particular model of consumer behavior described in (Sairamesh and Kephart, 1998), one can show analytically that in the limit of infinitely many consumers, the instantaneous profits per consumer Π_1 and Π_2 obtained by seller 1 and seller 2 respectively are given by:

$$\Pi_1 = \begin{cases} (Q_1 - p_1)(p_1 - c(Q_1)) & \text{if } 0 \leq p_1 \leq p_2 \text{ or } p_1 > Q_2 \\ (Q_1 - Q_2)(p_1 - c(Q_1)) & \text{if } p_2 < p_1 < Q_2 \end{cases} \quad (1)$$

$$\Pi_2 = \begin{cases} (Q_2 - p_2)(p_2 - c(Q_2)) & \text{if } 0 \leq p_2 < p_1 \\ 0 & \text{if } p_2 \geq p_1 \end{cases} \quad (2)$$

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 $c(Q) = 0.1(1 + Q)$. 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 $x_2 = Q_2 + c(Q_2)$):

$$p_1^*(p_2) = \begin{cases} Q_2 & \text{if } 0 \leq p_2 < x_1 - Q_2 \\ p_2 & \text{if } x_1 - Q_2 \leq p_2 \leq \frac{1}{2}x_1 \\ \frac{1}{2}x_1 & \text{if } p_2 > \frac{1}{2}x_1 \end{cases} \quad (3)$$

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:

$$p_2^*(p_1) = \begin{cases} c(Q_2) & 0 \leq p_1 \leq c(Q_2) \\ p_1 - \epsilon & \text{if } c(Q_2) \leq p_1 \leq \frac{1}{2}x_2 \\ \frac{1}{2}x_2 & \text{if } p_1 > \frac{1}{2}x_2 \end{cases} \quad (4)$$

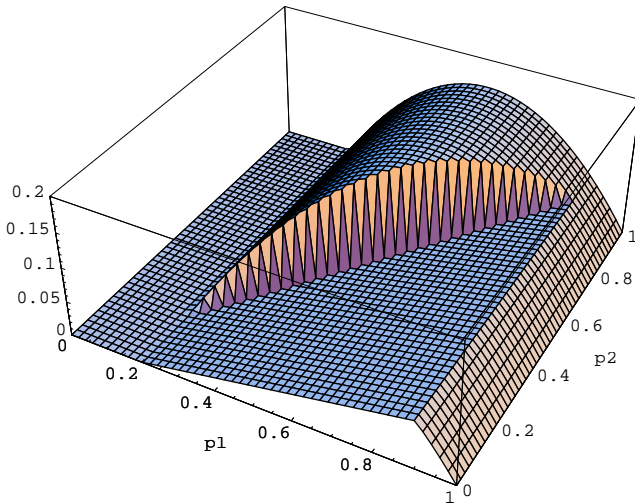


Figure 1: Sample profitability landscape for seller 1 in price-quality model, as a function of seller 1 price p_1 and seller 2 price p_2 .

We also note in passing that there are similar utility landscapes for each of the sellers in the information-filtering model (Kephart et al., 1998). In both 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 utility

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 utilities 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 a final simplification, we constrain the prices set by the two sellers to lie in a range from some minimum to maximum allowable price. The prices are also discretized, so that one can create lookup tables for the seller utility functions $\Pi(p_1, p_2)$. 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 Generalized minimax search

Recall that in two-player zero-sum games such as chess, minimax search to a fixed depth d works as follows: for a given starting position, one constructs a game tree of all possible moves, replies, counter-replies, etc., out to some fixed depth d . One then applies a heuristic evaluation function to the leaves of the tree, and one then does a minimax back-up of the values of the leaf nodes. This is done progressively starting from the bottom of the tree and working upwards, at each step applying either a min operation or a max operation depending on which side is moving. Another way of viewing this is that at depth 1 away from the leaves, the moves are selected based on a 1-ply search; at depth 2 away from the leaves, the selection is based on a 2-ply search; and so on, until finally at the root of the tree, the move that is selected is based on a d -ply search.

Our price-setting algorithm is analogous to this and works by building up a succession of optimal price lookup tables for successively greater depths. We begin by constructing a depth 1 optimal price table: for every possible starting price of the other seller, this table represents the best possible price that the seller can set to maximize immediate utility at the current time step. This corresponds to the myopic optimal pricing algorithm mentioned in the previous section, which was studied in detail in (Kephart et al., 1998). Since the state space is discretized and small, and since the consumer behavior is assumed to be a deterministic function of the prices of the two sellers, the optimal prices can be computed once for every state in the state space and stored in a table.

The next step is to construct a depth 2 optimal price table, which maximizes total utility summed over 2 time steps, assuming that the opponent will reply with a depth 1 optimal price. Next we construct a depth 3 table, which maximizes total utility summed over 3 time steps, assuming that the opponent will reply with a depth 2 price, and that the player will reply to that with a depth 1 price. We can continue in this way to generate optimal price tables of arbitrary depth. The optimal price table at depth n maximizes utility summed over n time steps, assuming that the opponent first responds with a $(n - 1)$ -step optimal price, the player then responds to that with a $(n - 2)$ -step optimal price, etc.. Optimal price tables can also be generated assuming discounting of future utilities governed by a discount parameter γ lying between 0 and 1. In this case the total discounted utility is maximized, where the predicted utility at m time steps in the future is weighted by a multiplicative factor of γ^m .

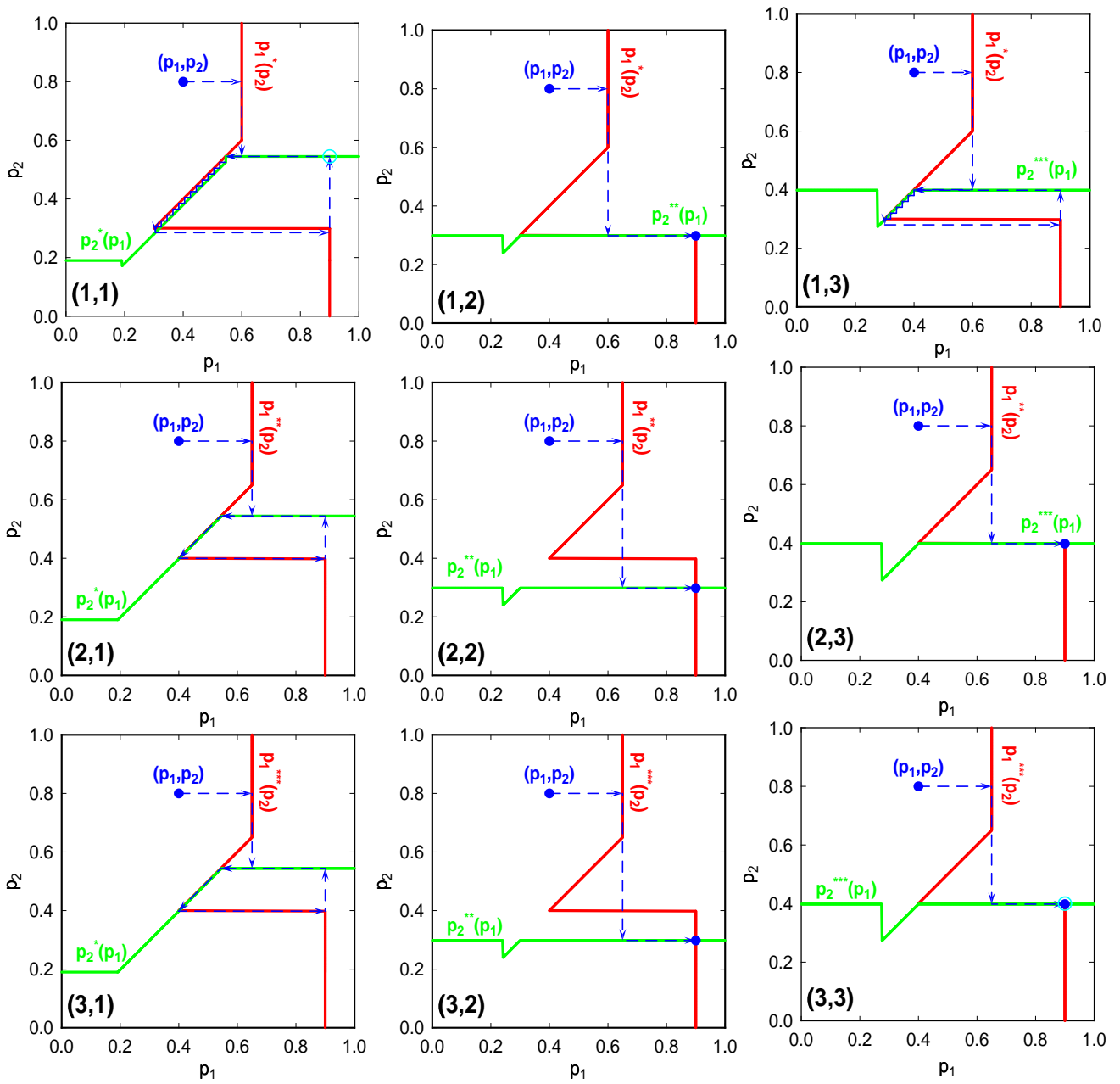


Figure 2: Plot of optimal price curves for seller 1 vs. seller 2 in the price-quality model at various lookahead depths. The numbers in parentheses in the lower left-hand corner of each plot indicate search depths (d_1, d_2) used by seller 1 and seller 2 respectively.

Of course, we don't expect the n -step trajectory predicted by this procedure to match the actual trajectory at every step. The later steps of the predicted trajectory in particular are based on smaller lookahead and therefore ought to be less accurate in matching agents using deep lookahead. However, the hope is that the first predicted step ought to give something reasonable. This is what has been found in domains such as chess — Deep Blue's predicted principal variations of 20-30 moves are generally not matched in exact detail, but its top level move decisions are nonetheless extremely accurate and strong.

Our basic finding is that when agents use any amount of lookahead at all, it can be sufficient to substantially curtail or eliminate the price war dynamics that result from myopic optimal pricing. An example of this, taken from the price-quality model, is shown in figure 2. In this figure, we have computed the optimal prices at lookahead depths 1, 2 and 3 for seller 1 ($p_1^*, p_1^{**}, p_1^{***}$) and seller 2 ($p_2^*, p_2^{**}, p_2^{***}$) and have systemically plotted each curve for seller 1 against each curve for seller 2. The lookahead depth is indicated in parentheses in the lower left-hand corner of each plot.

In each of these plots, the system dynamics for the state (p_1, p_2) can be obtained by alternately applying the two optimal price curves. 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. For example, one can see in the upper-left plot that when both sellers behave myopically, i.e. lookahead depths (1,1), the iterative graphical construction leads to a never-ending cyclic price war, whose trajectory is indicated by the dashed line. In contrast, when both sellers use either 2-step or 3-step lookahead, we can see that the price war is eliminated, and that the system dynamics instead iterates to a fixed point. The case when one of the sellers is myopic and the other seller uses greater lookahead is interesting. This is shown in the topmost three plots, where seller 1 is myopic, and in the leftmost three plots, where seller 2 is myopic. In the latter case, we see that when seller 2 is myopic, the price war cycle still exists but has a diminished amplitude. In contrast, when seller 1 is myopic, we see that if seller 2 uses 2-step lookahead, i.e. seller 2 has a perfect model of seller 1, the price war is eliminated. However, if seller 2 uses 3-step lookahead, the price war re-emerges, but at a diminished amplitude. The re-emergence of the price war comes about because, even though seller 2 is looking ahead further, it is now using an incorrect model of seller 1.

The locations of the fixed points shown in figure 2 are also of some interest, and can be simply explained. We note that in every case where a fixed point is obtained, the price for seller 1 is given by $p_1 = 0.9$, and that for the middle column of figure 2 (corresponding to seller 2 using 2-step lookahead) the price for seller 2 is $p_2 = 0.3$, whereas in the rightmost column (where seller 2 uses 3-step lookahead), seller 2's price is instead $p_2 = 0.4$. This can be simply understood in terms of seller 2's modeling of seller 1's behavior. When seller 2 uses 2-step lookahead, it models seller 1 as being myopic. The choice of $p_2 = 0.3$ represents the highest possible price at which seller 1 will not respond myopically by undercutting. This can be seen by inspection of the profit function given in equation 1; we can see that the profit in a single time step for seller 1 is 0.07 regardless of whether $p_1 = 0.9$ or $p_1 = 0.3$. On the other hand, when seller 2 uses 3-step lookahead, it is modeling seller 1 as using 2-step lookahead. The resulting price for seller 2, $p_2 = 0.4$, corresponds to the

point of indifference to undercutting for seller 1 based on a two-step optimization. If seller 1 does not undercut, he receives a profit of 0.07 for the next two time steps, for a total profit of 0.14. If seller 1 undercuts, he receives a profit of 0.12 on the first time step, followed on the next time step by a profit only 0.02 (after seller 2 retaliates with a further undercut), again resulting in a total return over two time steps of exactly 0.14. Thus the calculated fixed-point price for seller 2 has shifted to a higher value with greater lookahead. We note that these fixed points are at a lower price for seller 2 than the equilibrium point calculated in (Sairamesh and Kephart, 1998) for the one-shot (non-iterated) game. For the specific parameter settings used here, the calculated equilibrium values are $p_1 = 0.9$ and $p_2 = 0.545$. (This point corresponds to the open circle in the upper-left plot of figure 2. In the iterated game, this point is not a stable equilibrium when the players are myopic, because seller 1 can obtain a higher short-term reward by undercutting seller 2, and will therefore initiate a price war.)

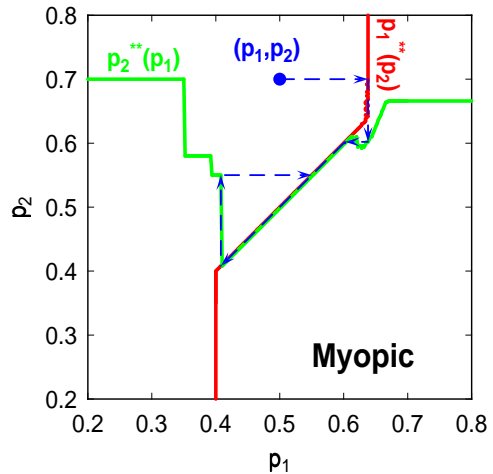


Figure 3: Plot of myopic optimal price curves for seller 1 vs. seller 2 in a sample run of the information-filtering model. The utilities for each seller were generated numerically using a simulated population of 250,000 consumers. Repeated iteration of the optimal price rules leads to a cyclic price war, as indicated by the dashed trajectory.

It is also of interest to examine lookahead depths greater than 3. One might hope that the optimal price curves converge to a unique invariant pair in the limit of large depth, and that the fixed point of the system would approach a Nash equilibrium. While we have no general proof that this will always happen, we have found empirically that, in the price-quality model, the optimal price curves for depths 4 and above turn out to be identical to the depth 3 curves.

However, we did not find this to be the case in the information-filtering model. Instead, it was found that there is a set of optimal price curves that periodically repeat with increasing depth, and that the period appears to be somewhat arbitrary, and can vary depending on the exact values of various cost parameters, etc., in the simulation. These findings were obtained regardless of whether or not discounting is used in the optimized utilities.

An example of results obtained in the information-filtering model is illustrated in figures 3 and 4. These results were

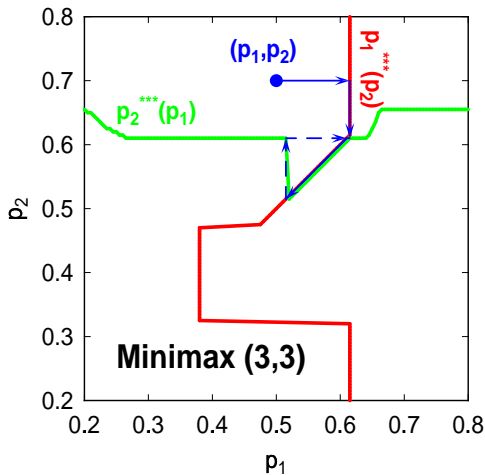


Figure 4: Plot of optimal price curves for seller 1 vs. seller 2, each using 3-step lookahead, based on the same information-filtering model used in figure 3. These optimal price curves still lead to price-war behavior, but with diminished amplitude.

obtained for a specific set of seller utilities that were numerically generated using a model population of 250,000 consumers. Figure 3 shows the myopic optimal price curves, while figure 4 shows the optimal price curves using 3-step lookahead. We have generally found in the information-filtering model that, for lookaheads depths of 2 or more, application of the calculated optimal curves resulted in a system dynamics in which the price-war cycles were either eliminated completely, or reduced in amplitude. This can be seen in figure 4, where the amplitude of the price-war cycle is reduced compared to that of figure 3.

In summary, we have shown that in two different models, the use of pricing algorithms based on n -step lookahead either eliminated or diminished the impact of price-war behavior which is obtained when both sellers behave myopically. Important issues for ongoing and future research include: (1) establishing under what conditions the n -step lookahead procedure converges in the limit of large n to a stationary set of optimal price curves; (2) understanding when the implied dynamics for a given pair of optimal price curves iterates to a fixed point, and when it yields limit-cycle behavior; (3) in those cases in which a fixed point is obtained, whether it corresponds to a Nash equilibrium of the system.

4 Generalized DP algorithms

While generalized minimax search is an efficient procedure for successively generating optimal price tables of greater and greater depth, it suffers from some theoretical inconsistencies. First, the generation of an n -step optimal price table for a given seller is based on the assumption that the opponent will be using an $(n - 1)$ -step policy. This is problematic if one believes that the opponent is using a search algorithm of equal depth. Furthermore, there is no way that both sellers could be correct in assuming that the opponent is using lesser-depth search. Second, there is the problem that successive steps in the predicted trajectory are based on progressively shallower depth search, until finally at the

end of the trajectory, the predicted pricing is myopic. Such predicted trajectories are unlikely to match actual trajectories, as it seems reasonable to assume that agents will use a given fixed depth search every time they are called upon to set a price.

As a way of overcoming these potential problems, we suggest that generalizing the formalism of Dynamic Programming from single-agent Markov decision problems to two-player arbitrary-sum games is a promising area of research. Such an approach could lead to the development of algorithms that yield optimal policies for both players that are fully accurate and self-consistent. Furthermore, DP-like approaches can be extended to large state spaces in which it is not feasible to use lookup tables for all possible states in the state space. This has been shown by numerous works in the field of Reinforcement Learning, in which the lookup tables of DP are replaced by compact state representations and nonlinear function approximators, and the full sweeps through the state space of DP are replaced by following actual trajectories generated by agent policies.

Within the context of the price-quality and information-filtering models mentioned previously, we have investigated a number of heuristic generalizations of DP, which attempt to do simultaneous, self-consistent optimization of the optimal price curves for both sellers. So far we have only examined the case of two-step lookahead, i.e., each seller optimizes two-step utility and also models the other seller as optimizing two-step utility. Within this context, we have examined a simple alternating policy iteration approach in which we start with initial curves $p_1(p_2)$ and $p_2(p_1)$ and alternately optimize p_1 and p_2 assuming that the other one is held fixed. Empirically, this approach was not found to converge to a unique, self-consistent solution. We have also looked at an asynchronous stochastic version of the above algorithm, in which one element at a time is chosen at random from each optimal price table, and only that element is optimized. This approach was not found to converge, either.

However, we have found empirical convergence with an approach based on an incremental version of DP, in which the prices are moved towards the computed optimal prices by small amounts. (This somewhat resembles a gradient-style minimization of the amount of inconsistency between the two optimal price curves at each time step.) When the asynchronous, stochastic algorithm mentioned above was combined with incremental price adjustments, it quickly converged in the price-quality model to the exact same curves (apart from very small random fluctuations of less than 1%) as obtained by 3 or more lookahead steps in the generalized minimax procedure. A plot of the DP-generated optimal price curves is shown below in figure 5. We note that visually this plot appears identical to the lower right-hand plot in figure 2, and leads to fixed-point dynamics rather than cyclic price wars.

The same stochastic, asynchronous, incremental DP algorithm was tested in the information-filtering model, and when the number of consumers in the simulation was large (10,000 to 250,000) it gave similarly good approximate convergence within small fluctuations to stationary optimal price curves. However, with only 1000 consumers in the model, the fluctuations were larger and it was not entirely clear whether the algorithm was converging to a stationary solution. We conjecture that this behavior may be due to differing degrees of smoothness in the utility landscapes. The analytically-defined landscapes in the price-quality model are very smooth, and are also smooth in the filtering model as the number of consumers becomes infinitely large. How-

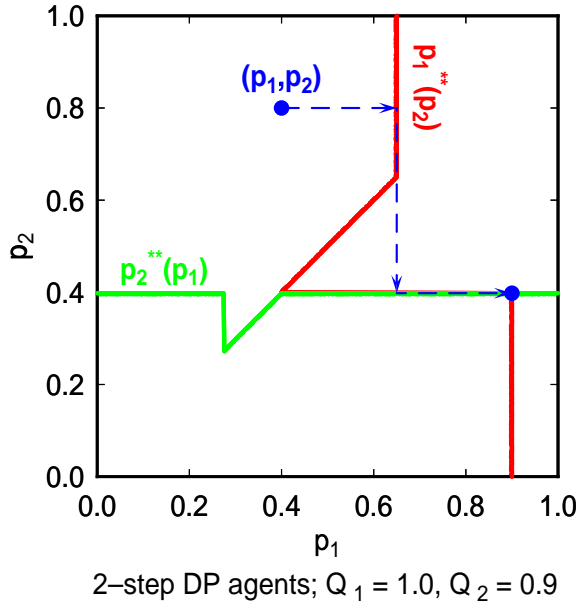


Figure 5: Two-step optimal prices for seller 1 vs. seller 2 calculated by stochastic, asynchronous, incremental DP in price-quality model.

ever, the landscapes become much more ragged with a small finite-size consumer population, and such raggedness may limit the ability of the DP-like approach to converge to a stationary solution.

An example of results using the DP-like approach in the information-filtering model is shown in figure 6. This figure plots the two-step optimal price curves for seller 1 and seller 2 for the same numerically generated utilities as were used in figures 3 and 4. We do find good approximate convergence to stationary curves in this model. The small random fluctuations are somewhat larger than those seen in figure 5; most likely this is due to a coarser resolution in the optimal price tables (.005 vs. .002). In this particular model, we can see that the DP-generated optimal price curves still lead to a price war that actually has a slightly larger amplitude than in the myopic case shown previously in figure 3. Hence there are at least some cases in which the use of lookahead in the sellers' pricing strategy can actually exacerbate the price-war dynamics. Determining under what conditions price wars are amplified by lookahead, and under what conditions they are diminished or eliminated, is an important topic for further research.

5 Conclusions

We have constructed a very simplified and restricted two-seller economy in which the instantaneous utilities of the two sellers are given either by the price-quality model of (Sairamesh and Kephart, 1998), or by the of the information-filtering model of (Kephart et al., 1998). In both models, cyclic price wars are obtained when the sellers myopically optimize their instantaneous utilities without regard to longer-term impact of their pricing policies. By limiting the system to two sellers with fixed product differentiation (i.e. quality or information category), and by modeling the con-

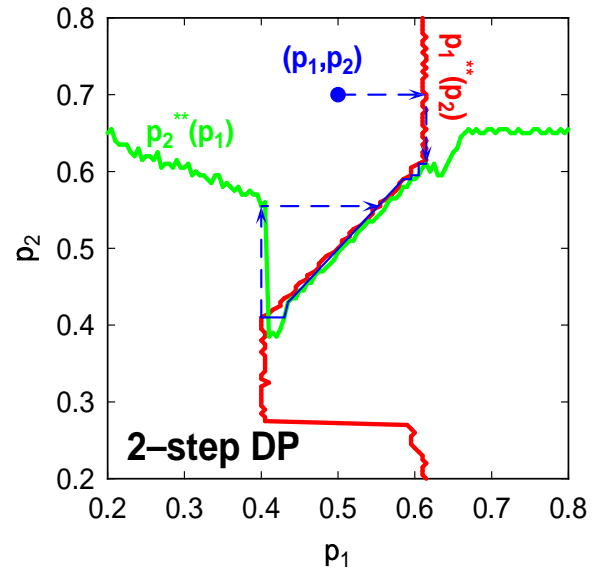


Figure 6: Two-step optimal prices for seller 1 vs. seller 2 calculated by stochastic, asynchronous, incremental DP in the same information-filtering model as shown in figures 3 and 4.

sumers as giving instantaneous, deterministic responses to the current prices of the two sellers, we have essentially created a two-player, alternating-turn, arbitrary-sum Markov game. In this game, both the state-space transitions and the rewards at each time step are deterministic; furthermore, the state space is fully observable and the exact utility functions for both players are known by each player. Finally, the discretization of the seller prices implies that lookup table representation of the utilities and optimal prices are feasible.

Under the above conditions, we have developed two types of procedures for progressively calculating optimal price tables based on n -step lookahead. Our main finding is that, at least for the two specific models studied here, it was possible to significantly ameliorate the myopic price-war dynamics without resorting to deep searches. In both models, utilizing lookahead depths as small as $n = 2$ was sufficient in at least some cases to damp out or eliminate price wars. These results confirmed our intuition that the ability to anticipate retaliatory responses can be an important mechanism in avoiding price wars, and provide some degree of hope that similar findings might be obtained in more realistic scenarios of larger numbers of agents, and less than perfect information available to the agents. (If foresight-based algorithms turned out not to work in the simplest case, there would be no reason to expect them to work in more realistic cases.) It will be of great interest to understand the conditions under which this will be the case, and when mutual prediction of any depth will have little or no impact on systemic price wars, or possibly even make them worse.

Within the context of our current simulations, one important area of ongoing and future research is further elaborating the conditions under which the proposed algorithms converge to invariant optimal price curves. Several important challenges will also be faced in extending the algorithms for agent foresight to larger-scale, more realistic simulations. In the work reported here, the state space was small and

perfectly known, so that one could hope to develop an algorithm that could calculate in advance something like a game-theoretic optimal pricing algorithm for each agent. When there are a large number of sellers in the simulation, the seller utilities 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. 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 lookahead procedure has to anticipate all possible orderings of opponent responses. Finally, although the sellers may know quite a lot about the other sellers' costs and rewards, it seems unrealistic to hope that sellers could have full knowledge of the other sellers' utility functions.

A further set of issues that would be of interest to address has to do with what agents might do when they observe that the actual behavior of other agents does not match with predicted behavior. One could argue that this might not be of paramount importance. As long as the predicted behaviors lead to the elimination of price wars, perhaps any further refinement in prediction accuracy would not lead to significantly increased rewards. Nevertheless, if one observes other agents behaving in a way that one believes to be suboptimal, it might be worth investigating whether such apparent suboptimal behavior might possibly be exploited by some sort of inductive modification of the agent's predictions. A number of important theoretical challenges must be faced in trying to develop such algorithms. First, there is the issue of generalization: given that we have observed the opponent behaving in a certain way at one point in state space, what can we then infer about how that opponent will behave in other parts of the state space? Second, there is the issue of exploration: it may be necessary for the agent to make suboptimal moves in order to reach new areas of state space, simply to gather more information about the opponents' behaviors and strategies. Third, there are additional challenges if one believes that the opponents' strategies are non-stationary; this is presumably much more difficult than modeling a fixed strategy. Finally, there are a host of gamemanship-type issues that could arise; for example, if we observe an opponent behaving suboptimally, it may be a trick — the opponent may be trying to lead us to develop an incorrect model which can then be exploited at later times.

Acknowledgements

The authors thank J. Sairamesh and Amy Greenwald for helpful discussions.

References

- [1] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Belmont, MA: Athena Scientific, 1995.
- [2] D. Foster and R. Vohra, "Regret in the on-line decision problem." *Games and Economic Behavior*, to appear, 1998.
- [3] J. Hu and M. P. Wellman, "Self-fulfilling bias in multi-agent learning," Proceedings of ICMAS-96, AAAI Press, 1996.
- [4] J. O. Kephart, J. E. Hanson and J. Sairamesh, "Pricer dynamics in a free-market economy of software

agents," to appear in: Proceedings of ALIFE-VI, Los Angeles, 1998.

- [5] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," Proceedings of the Eleventh International Conference on Machine Learning, 157-163, Morgan Kaufmann, 1994.
- [6] P. Milgrom and J. Roberts. "Adaptive and sophisticated learning in normal form games," *Games and Economic Behavior*, 3:82-100, 1991.
- [7] J. Sairamesh and J. O. Kephart, "Dynamics of price and quality differentiation in information and computational markets," submitted to ICE-98, 1998.
- [8] J. M. Vidal and E. H. Durfee, "Learning nested agent models in an information economy," *J. of Experimental and Theoretical AI*, to appear, 1998.