A Learning Algorithm for Agents in Electronic Marketplaces

Thomas Tran and **Robin Cohen**
Department of Computer Science
University of Waterloo
Waterloo, ON, Canada N2L 3G1
{tt5tran, rcohen}@math.uwaterloo.ca

Abstract

In this paper, we propose a reputation oriented reinforcement learning algorithm for buying and selling agents in electronic market environments. We take into account the fact that multiple selling agents may offer the same good with different qualities. In our approach, buying agents learn to avoid the risk of purchasing low quality goods and to maximize their expected value of goods by dynamically maintaining sets of reputable sellers. Selling agents learn to maximize their expected profits by adjusting product prices and by optionally altering the quality of their goods. Modelling the reputation of sellers allows buying agents to focus on those sellers with whom a certain degree of trust has been established. We also include the ability for buying agents to optionally explore the marketplace in order to discover new reputable sellers. As detailed in the paper, we believe that our proposed strategy leads to improved satisfaction for buyers and sellers, reduced communication load, and robust systems. In addition, we outline some possible experimentation with an implementation of the algorithm, to determine its potential advantages.

Introduction

The problem of how to design personal, intelligent agents for e-commerce applications is a subject of increasing interest from both the academic and industrial research communities (Chavez & Maes 1996; Doorenbos, Etzioni, & Weld 1997; Wurman, Wellman, & Wash 1998). Since a multi-agent electronic market environment is, by its very nature, open (agents can enter or leave the environment at will), dynamic (information such as prices, product quality etc. may be altered), and unpredictable (agents lack perfect knowledge of one another), it is very important that participant agents are equipped with effective and feasible learning algorithms to accomplish their delegated tasks or achieve their delegated goals. In this paper, we propose a reinforcement learning and reputation based algorithm for buying and selling agents in electronic market environments.

We model the agent environment as an open marketplace which is populated with economic agents. The nature of an open marketplace allows economic agents, which we classify as *buyers* and *sellers*, to freely enter or leave the market. Buyers and sellers are self-interested agents whose goal is to maximize their own benefit. Buying and selling prices are determined by individual buyers and sellers respectively, based on their aggregate past experiences.

Our market environment is rooted in an information delivery infrastructure such as the Internet, which provides agents with virtually direct and free access to all other agents. The process of buying and selling products is realized via a *contract-net* like mechanism (Davis & Smith 1983; Smith 1980), which consists of three elementary phrases: *(i)* A buyer announces its desire for a good. *(ii)* Sellers submit bids for delivering such goods. *(iii)* The buyer evaluates the submitted bids and selects a suitable seller. The buyer then pays the chosen seller and receives the good from that seller. Thus, the buying and selling process can be viewed as an auction where a seller is said to be "winning the auction" if it is able to sell its good to the buyer.

We assume that the quality of a good offered by different sellers may not be the same, and a seller may alter the quality of its goods. We also assume that a buyer can examine the quality of the good it purchases only after it receives that good from the selected seller. Each buyer has some way to evaluate the good it purchases, based on the price and the quality of the good received. Thus, in our market environment a buyer tries to find those sellers whose goods best meet its expected value of goods, while a seller tries to maximize its expected profit by setting suitable prices for and providing more customized value to its goods, in order to satisfy the buyers' needs.

Reinforcement learning has been studied for various multi-agent problems (Littman 1994; Nagayuki, Ishii, & Doya 2000; Ono & Fukumoto 1996; Sandholm & Crites 1995; Sen, Sekaran, & Hale 1994; Weiss 1993). However, the agents and environments in these works are not directly modeled as economic agents and market environments. (Vidal & Durfee 1996) does apply reinforcement learning in market environments for buying and selling agents, but does not use reputation as a means to protect buyers from purchasing low quality goods. Moreover, selling agents in (Vidal

& Durfee 1996) do not consider altering the quality of their products while learning to maximize their profits.

In our proposed learning algorithm, buyers are designed to be reputation-oriented to avoid the risk of purchasing unsatisfactory quality goods. They each dynamically maintain a set of sellers with good reputation, and learn to maximize their expected product values by selecting appropriate sellers among those reputable sellers. Sellers in our approach learn to maximize their expected profits by not only adjusting product prices but also by optionally altering the quality of their products. As discussed in detail later, we believe that the proposed algorithm will result in improved performance for buyers, better satisfaction for both buyers and sellers, reduced communication load, and more robust systems.

The paper is organized as follows: The next section, section 2, introduces our proposed learning algorithm for buyers and sellers, respectively. This section also includes a numerical example that demonstrates how the proposed algorithm works, and a discussion of the possible advantages of the algorithm. Section 3 remarks on related work. Section 4 outlines the proposed experimentation with the model, and section 5 provides some future research directions. Finally, section 6 concludes the paper.

The Proposed Learning Algorithm

In this section we propose a reinforcement learning and reputation based algorithm for buyers and sellers, respectively. The algorithm is aimed at maximizing the expected values of goods for buyers, and maximizing the expected profits for sellers. Note that it is quite possible for both a seller s and a buyer b to be "winning" in a business transaction. This happens when seller s could choose a price p to sell good g to buyer b that maximized its expected profit, and buyer b decided that purchasing good g at price p from seller s would maximize its expected value of goods. We also provide a numerical example to illustrate how the algorithm works.

Buying Algorithm

Consider the scenario where a buyer b makes an announcement of its desire for some good g. Let G be the set of goods, P be the set of prices, and S be the set of all sellers in the marketplace. G, P, and S are finite sets.

Let S_r^b be the set of sellers with good reputation to buyer b; that is, S_r^b contains the sellers that have served b well in the past and are therefore trusted by b. Hence, $S_r^b \subseteq S$ and S_r^b is initially empty. To measure the reputation of a seller $s \in S$, buyer b uses a real-valued function $r^b : S \mapsto (-1, 1)$, which is called the *reputation function* of b. Initially, buyer b sets $r^b(s) = 0$ for all $s \in S$. Thus, the set S_r^b consists of all sellers s with $r^b(s) \geq \Theta > 0$, where Θ is the *reputation threshold* determined by b, i.e.,

$$S_r^b = \{s \in S \mid r^b(s) \geq \Theta > 0\} \subseteq S.$$

Buyer b estimates the expected value of the goods it purchases using the *expected value function* $f^b : G \times P \times S \mapsto \mathbb{R}$. Hence, the real number $f^b(g, p, s)$ represents buyer b's expected value of buying good g at price p from seller s.

Since a seller may alter the quality of its goods, buyer b puts more trust in the sellers with good reputation. Thus, it chooses among the reputable sellers in S_r^b a seller $\hat{s}$ that offers good g at price p with maximum expected value:

$$\hat{s} = \arg \max_{s \in S_r^b} f^b(g, p, s), \tag{1}$$

where arg is an operator such that arg $f^b(g, p, s)$ returns s.

If no sellers in S_r^b submit bids for delivering g (or if $S_r^b = \emptyset$), then buyer b will have to choose a seller $\hat{s}$ from the set of non-reputable sellers:

$$\hat{s} = \arg \max_{s \in (S - S_r^b)} f^b(g, p, s). \tag{2}$$

In addition, with a probability ρ, buyer b chooses to explore (rather than exploit) the marketplace by randomly selecting a seller $\hat{s}$ from the set of all sellers. This gives buyer b an opportunity to discover new reputable sellers. Initially, the value of ρ should be set to 1, then decreased over time to some fixed minimum value determined by the buyer.

After paying seller $\hat{s}$ and receiving good g, buyer b can examine the quality $q \in Q$ of good g, where Q is a finite set of real values representing product qualities. It then calculates the true value of good g using the function $v^b : P \times Q \mapsto \mathbb{R}$. For instance, if $p = q$ and buyer b prefers the sellers that offer goods with higher quality, it may set $v^b(p, q) = cq - p$, where c is a constant greater than 1.

The expected value function f^b is now incrementally learned in a reinforcement learning framework:

$$\Delta = v^b(p, q) - f^b(g, p, \hat{s}), \tag{3}$$

$$f^b(g, p, \hat{s}) \leftarrow f^b(g, p, \hat{s}) + \alpha\Delta, \tag{4}$$

where α is called the *learning rate* ($0 \leq \alpha \leq 1$). The learning rate should be initially set to a starting value of 1 and, similar to ρ, be reduced over time to a fixed minimum value chosen depending on individual buyers.

Thus, if $\Delta = v^b(p, q) - f^b(g, p, \hat{s}) \geq 0$ then $f^b(g, p, \hat{s})$ is updated with the same or a greater value than before. This means that seller $\hat{s}$ has a chance to be chosen by buyer b again if it continues offering good g at price p in the next auction. Conversely, if $\Delta < 0$ then $f^b(g, p, \hat{s})$ is updated with a smaller value than before. So, seller $\hat{s}$ may not be selected by buyer b in the next auction if it continues selling good g at price p.

In addition to updating the expected value function, the reputation rating $r^b(\hat{s})$ of seller $\hat{s}$ also needs to be updated. Let $\vartheta^b(g) \in \mathbb{R}$ be the product value that buyer b demands for good g. We use a reputation updating scheme motivated by that proposed in (Yu & Singh 2000), as follows:

If $\delta = v^b(p,q) - \vartheta^b(g) \geq 0$, that is, if seller $\hat{s}$ offers good g with value greater than or equal to the value demanded by buyer b, then its reputation rating $r^b(\hat{s})$ is increased by

$$r^b(\hat{s}) \leftarrow \begin{cases} r^b(\hat{s}) + \mu(1 - r^b(\hat{s})) & \text{if } r^b(\hat{s}) \geq 0, \\ r^b(\hat{s}) + \mu(1 + r^b(\hat{s})) & \text{if } r^b(\hat{s}) < 0, \end{cases} \tag{5}$$

where μ is a positive constant called the *cooperation factor*[1] ($0 < \mu < 1$).

Otherwise, if $\delta < 0$, that is, if seller $\hat{s}$ sells good g with value less than that demanded by buyer b, then its reputation rating $r^b(\hat{s})$ is decreased by

$$r^b(\hat{s}) \leftarrow \begin{cases} r^b(\hat{s}) + \nu(1 - r^b(\hat{s})) & \text{if } r^b(\hat{s}) \geq 0, \\ r^b(\hat{s}) + \nu(1 + r^b(\hat{s})) & \text{if } r^b(\hat{s}) < 0, \end{cases} \tag{6}$$

where ν is a negative constant called the *non-cooperation factor* ($-1 < \nu < 0$).

To protect itself from dishonest sellers, buyer b may require that $|\nu| > |\mu|$. This implements the traditional idea that reputation should be difficult to build up, but easy to tear down.

The set of reputable sellers to buyer b now needs to be updated based on the new reputation rating $r^b(\hat{s})$, as in one of the following two cases:

(i) If ($\hat{s} \in S_r^b$) and ($r^b(\hat{s}) < \Theta$) then buyer b no longer considers $\hat{s}$ as a reputable seller, i.e.,

$$S_r^b \leftarrow S_r^b - \{\hat{s}\}. \tag{7}$$

(ii) If ($\hat{s} \notin S_r^b$) and ($r^b(\hat{s}) \geq \Theta$) then buyer b now considers $\hat{s}$ as a seller with good reputation, i.e.,

$$S_r^b \leftarrow S_r^b \cup \{\hat{s}\}. \tag{8}$$

Let us now look at the sellers' algorithm.

Selling Algorithm

Consider the scenario where a seller $s \in S$ has to decide on the price to sell some good g to a buyer b. Let B be the (finite) set of buyers in the marketplace and let function $h^s : G \times P \times B \mapsto \mathbb{R}$ estimate the expected profit for seller s. Thus, the real number $h^s(g,p,b)$ represents the expected profit for seller s if it sells good g at price p to buyer b. Let $c^s(g,b)$ be the cost of seller s to produce good g for buyer b. Note that seller s may produce various versions of good g, which are tailored to meet the needs of different buyers. Seller s will choose a price $\hat{p}$ greater than or equal to cost $c^s(g,b)$ to sell good g to buyer b such that its expected profit is maximized:

$$\hat{p} = \arg \max_{\substack{p \in P \\ p \geq c^s(g,b)}} h^s(g,p,b), \tag{9}$$

where in this case arg is an operator such that $\arg h^s(g,p,b)$ returns p.

[1]Buyer b will consider seller $\hat{s}$ as being *cooperative* if the good $\hat{s}$ sells to b has value greater than or equal to that demanded by b.

The expected profit function h^s is learned incrementally using reinforcement learning:

$$h^s(g,p,b) \leftarrow h^s(g,p,b) + \alpha(Profit^s(g,p,b) - h^s(g,p,b)), \tag{10}$$

where $Profit^s(g,p,b)$ is the actual profit of seller s if it sells good g at price p to buyer b.

Function $Profit^s(g,p,b)$ is defined as follows:

$$Profit^s(g,p,b) = \begin{cases} p - c^s(g,b) & \text{if } s \text{ wins the auction,} \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

Thus, if seller s does not win the auction then $(Profit^s(g,p,b) - h^s(g,p,b))$ is negative, and by (10), $h^s(g,p,b)$ is updated with a smaller value than before. This reduces the chance that price $\hat{p}$ will be chosen again to sell good g to buyer b in future auctions. Conversely, if seller s wins the auction then price $\hat{p}$ will probably be re-selected in future auctions.

If seller s succeeded in selling good g to buyer b once, but subsequently fails for a number of auctions, say for m consecutive auctions (where m is seller s specific constant), then it may not only because s has set a too high price for good g, but probably also because the quality of g does not meet buyer b's expectation. Thus, in addition to lowering the price via equation (10), seller s may optionally add more value (quality) to g by increasing its production cost[2]:

$$c^s(g,b) \leftarrow (1 + Inc)c^s(g,b), \tag{12}$$

where Inc is seller s specific constant called the *quality increasing factor*.

In contrast, if seller s is successful in selling good g to buyer b for n consecutive auctions, it may optionally reduce the quality of good g, and thus try to further increase its future profit:

$$c^s(g,b) \leftarrow (1 - Dec)c^s(g,b), \tag{13}$$

where Dec is seller s specific constant called the *quality decreasing factor*.

An Example

This subsection provides a numerical example illustrating the proposed algorithm for buyers and sellers, respectively.

Buying Situation

Consider a simple buying situation where a buyer b announces its need of some good g. Suppose that there are 6 sellers in the marketplace, i.e.,

$$S = \{s_i \mid i = 1..6\},$$

and that the set of sellers with good reputation to b is

$$S_r^b = \{s_j \mid j = 1..3\} \subset S.$$

[2]This supports the common assumption that high quality goods cost more to produce.

Furthermore, suppose $\Theta = 0.4$, $v^b(p,q) = 2.5q - p$, $\alpha = 0.8$, $\vartheta^b(g) = 6.10$, $\mu = 0.2$, $\nu = -0.4$, and the reputation ratings $r^b(s_i)$ are given as follows:

s_i	s_1	s_2	s_3	s_4	s_5	s_6
$r^b(s_i)$	0.40	0.45	0.50	0.30	0.25	0.20

Table 1: Reputation ratings of different sellers to buyer b.

After b's announcement of its desire for g, the sellers bid with the following prices to deliver g to b:

s_i	s_1	s_2	s_3	s_4	s_5	s_6
p	4	5	4.5	4	5	3.5

Table 2: Prices offered by different sellers for good g.

Assume that b's expected values of buying g at various prices from different sellers are

s_i	s_1	s_2	s_3	s_4	s_5	s_6
p	4	5	4.5	4	5	3.5
$f^b(g,p,s_i)$	6.15	7.25	6.65	5.50	5.75	5.20

Table 3: Buyer b's expected value of buying good g at various prices from different sellers.

Then, by equation (1), b buys g from s_2 at price $p = 5$ with

$$f^b(g,p,s_2) = 7.25 = \max_{s \in S_r^b} f^b(g,p,s).$$

Suppose b examines the quality q of good g and finds that $q = 5$. It then calculates the true value of g:

$$v^b(p,q) = 2.5q - p = 2.5(5) - 5 = 7.50.$$

Buyer b now updates its expected value function using equations (3) and (4):

$$\begin{aligned}\Delta &= v^b(p,q) - f^b(g,p,s_2)\\ &= 7.50 - 7.25 = 0.25, \text{ and}\\ f^b(g,p,s_2) &\leftarrow f^b(g,p,s_2) + \alpha\Delta\\ &\leftarrow 7.25 + (0.80)(0.25) = 7.45.\end{aligned}$$

Finally, since $\delta = v^b(p,q) - \vartheta^b(g) = 7.50 - 6.10 \geq 0$, buyer b increases the reputation rating $r^b(s_2)$ of seller s_2 according equation (5):

$$\begin{aligned}r^b(s_2) &\leftarrow r^b(s_2) + \mu(1 - r^b(s_2))\\ &\leftarrow 0.45 + (0.20)(1 - 0.45) = 0.56.\end{aligned}$$

Thus, by providing good g with high value, seller s_2 has improved its reputation to buyer b and remained in the set S_r^b of reputable sellers to b.

Selling Situation

Consider how a seller in the above-said marketplace, say seller s_4, behaves according to the proposed selling algorithm. Suppose $c^{s_4}(g,b) = 2.5$, $\alpha = 0.8$, and $Inc = Dec = 0.1$.

Upon receiving buyer b's announcement of its desire for good g, s_4 has to decide on the price to sell g to b. Assume that s_4's expected profits to sell g to b at various prices are

p	2.5	2.75	3.0	3.25	3.5	3.75	4.0	4.25	4.5
h^{s_4}	0.0	0.25	0.5	0.75	1.0	1.25	1.5	0.0	0.0

Table 4: Expected profits of seller s_4 in selling good g to buyer b at different prices.

Table 4 indicates that s_4 does not expect to be able to sell g to b at price $p \geq 4.25$. By equation (9), s_4 chooses price $\hat{p} = 4$ to sell g to b:

$$\hat{p} = \arg \max_{\substack{p \in P \\ p \geq c^s(g,b)}} h^{s_4}(g,p,b) = 4.$$

Since b chooses to buy g from another seller, namely s_2, the actual profit of s_4 is zero, i.e., $Profit^{s_4}(g,\hat{p},b) = 0$. Hence, s_4 updates its expected profit using equation (10) as follows:

$$\begin{aligned}h^{s_4}(g,\hat{p},b) &\leftarrow h^{s_4}(g,\hat{p},b) + \alpha(Profit^{s_4}(g,\hat{p},b) - h^{s_4}(g,\hat{p},b))\\ &\leftarrow 1.50 + (0.80)(0 - 1.50) = 0.30.\end{aligned}$$

Thus, according to equation (9), it is unlikely that price $\hat{p} = 4$ will be chosen again to sell good g to buyer b in future auctions.

Assume that s_4 has failed to sell g to b for a number of auctions. It therefore decides to add more quality to g by increasing the production cost using equation (12):

$$\begin{aligned}c^s(g,b) &\leftarrow (1 + Inc)c^s(g,b)\\ &\leftarrow (1 + 0.10)(2.5) = 2.75.\end{aligned}$$

By doing so, seller s_4 hopes that good g may now meet buyer b's quality expectation and that it will be able to sell g to b in future auctions.

Discussion

Work in the area of software agents has been focusing on how agents should cooperate to provide valuable services to one another. However, answering the question of why agents should cooperate with one another at all is also of equal importance (Kephart 2000). We believe that modeling an agent environment as a marketplace, where agents are motivated by economic incentives to provide goods and services to each other, is a practical and feasible approach.

It is possible in the marketplace that some sellers may try to "cheat" by delivering some good with reasonable quality, followed by low quality goods. Buyers in our approach use reputation as a means to protect themselves from those dishonest sellers: They each dynamically maintain a set of reputable sellers and consider choosing suitable sellers from this set first. This strategy reduces a buyer's risk of purchasing low quality goods, and therefore brings better satisfaction to the buyer.

Since a buyer's set of reputable sellers is certainly a lot smaller (in terms of cardinality) than the set of all sellers in the market, the proposed buying algorithm reduces computational cost, and accordingly results in improved performance for the buyer (compared to the case where the buyer has to consider all possible sellers). This is especially important in those application domains where the buyer is required to calculate a suitable seller within a constrained time frame. For instance, if the buyer serves some user as a personal assistant, then it must respond to the user within an allowable time period. We note that various buyers may use different reputation thresholds (thus, resulting in dissimilar sets of reputable sellers) as well as different learning rates.

In our proposed buying algorithm, a buyer selects a seller based on its own past experience and doesn't communicate with other buyers for its decision. We believe that this type of learning has certain advantages: Buyers can act independently and autonomously without being affected by communication delays (due to other buyers being busy), the failure of some key-buyer (whose buying policy influences other buyers), or the reliability of the information (the information received from other buyers may not be reliable). The resultant system, therefore, should be more robust (Sen, Sekaran, & Hale 1994).

The underlying mechanism that allows agents to do business with one another in our marketplace is actually a form of the contract-net protocol (Davis & Smith 1983; Smith 1980), where buyers announce their desire for goods to all sellers via multicast or possibly broadcast. This works well in small and moderate-sized environments; however, as the problem size (i.e., the number of communicating agents and the number of desired goods) increases, this may run into difficulties due to the slow and expensive communication. The proposed buying algorithm provides a potential solution to this problem: A buyer may just send announcements of its desire for goods to its reputable sellers instead of all sellers, and thus reducing the communication load and increasing the overall system performance.

Since the marketplace is open and sellers are continuously learning to improve their profits, some new, good sellers may have entered the market, and/or some non-reputable sellers may have reasonably adjusted their prices and greatly improved the quality of their products, and thus should be considered as reputable sellers. The proposed buying strategy accounts for this possibility by letting a buyer b explore the marketplace with probability ρ to discover new reputable sellers.

The proposed selling strategy is suitable for sellers in market environments where a seller can only sell its products and gain profit by winning auctions. There are two important reasons why a seller may not be able to win an auction in our market environment: *(i)* It may set the price too high, and *(ii)* the quality of its product may be under the buyer's expectation level. Our proposed selling algorithm considers both of these factors by allowing the seller to not only adjust the price (equation (10)), but also optionally add more quality to its product (equation (12)). Various sellers may have different policies for adjusting prices and altering the quality of their products. This is reflected by the way they choose their learning rates and how they increase/decrease their production costs, e.g., using linear functions as in equations (12) and (13), or using more sophisticated functions.

Related Work

Reinforcement learning has been studied in various multi-agent problems such as pursuit games (Ono & Fukumoto 1996), soccer (Littman 1994), the prisoner's dilemma game (Sandholm & Crites 1995), and coordination games (Sen, Sekaran, & Hale 1994). However, the agents and environments studied in these works are not economic agents and market environments. The reinforcement learning based algorithm proposed in this paper is, in contrast, aimed at application domains where agents are economically motivated and act in open market environments.

In addition, our work contrasts with other efforts to assist users in buying and selling goods in electronic marketplaces. A number of agent models for electronic market environments have been proposed. Jango (Doorenbos, Etzioni, & Weld 1997) is a shopping agent that assists customers in getting product information. Given a specific product by a customer, Jango simultaneously queries multiple online merchants (from a list maintained by NetBot, Inc.) for the product availability, price, and important product features. Jango then displays the query results to the customer. Although Jango provides customers with useful information for merchant comparison, at least three shortcomings may be identified: *(i)* The task of analyzing the resultant information and selecting appropriate merchants is completely left for customers. *(ii)* The algorithm underlying its operation does not consider product quality, which is of great importance for the merchant selection task. *(iii)* Jango is not equipped with any learning capability to help customers choose more and more appropriate merchants. Another interesting agent model is *Kasbah* (Chavez & Maes 1996), designed by the MIT Media Lab. Kasbah is a multi-agent electronic marketplace where selling and buying agents can negotiate with one another to find the "best possible deal" for their users. The main advantage of Kasbah is that its agents are autonomous in making decisions, thus freeing users from having to find and negotiate with buyers and sellers. However, as admitted in (Chavez & Maes 1996), Kasbah's agents are not very smart as they do not make use of any AI learning techniques.

Vidal and Durfee (Vidal & Durfee 1996) address the problem of how buying and selling agents should behave in an information economy such as the University of Michigan Digital Library. They divide agents into classes corresponding to the agents' capabilities of modeling other agents: Zero-level agents are the agents that learn from the observations they make about their environment, and from any environmental rewards they receive. One-level agents are those agents that model other agents as zero-level agents. Two-level agents are those that model other agents as one-level agents. Higher level agents are recursively defined in the same manner. It should be intuitive that the agents with more complete models of others will always do better. However, because of the computational costs associated with maintaining deeper (i.e., more complex) models, there should be a level at which the gains and the costs of having deeper models balance out for each agent. The main problem addressed in (Vidal & Durfee 1996) is to answer the question of when an agent benefits from having deeper models of others.

The work in (Vidal & Durfee 1996) motivates and serves as a starting point for our work. Nevertheless, we believe that in a market environment, reputation of sellers is an important factor that buyers can exploit to avoid interaction with dishonest sellers, therefore reducing the risk of purchasing low quality goods. On the other hand, we think that sellers may increase their sales (and hence their profits) by not only adjusting the prices of their goods, but also by tailoring their goods to meet the buyers' specific needs. Thus, instead of having agents maintain recursive models of others and dealing with the associated computational costs, we consider taking a new approach: We would like to use a reputation mechanism as a means of shielding buyers from being "cheated" (by malicious sellers), and to give sellers the option of altering the quality of their goods to satisfy the buyers' needs.

Proposed Experimentation

For the next step, we would like to experimentally confirm the possible advantages of the proposed algorithm. In fact, we are currently constructing a simulation of a marketplace populated with agents using the algorithm. The marketplace is implemented using Java 2 with the following initial set of parameters:

- The reputation threshold $\Theta = 0.4$.
- The true product value function $v^b(p, q) = 3.5q - p$.
- The cooperation factor $\mu = 0.07$, and the non-cooperation factor $\nu = -0.7$.
- The number of consecutive unsuccessful auctions (after which a seller may consider improving product quality by increasing production cost) $m = 10$, and the number of consecutive successful auctions (after which a seller may consider decreasing production cost for further profit) $n = 10$.
- The quality increasing factor $Inc = 0.05$, and the quality decreasing factor $Dec = 0.05$.
- The exploration probability ρ and the learning rate α are initially set to 1 and then decreased overtime (by factor 0.995) until they reach $\rho_{min} = 0.2$ and $\alpha_{min} = 0.2$.

We choose the quality q of a good to be equal to the cost for producing that good, in order to support the common idea that it costs more to produce high quality goods. The simulation is set up in such a way that the above parameters can be changed to explore agent behaviours in different test sets. We would initially choose a modest sized marketplace, for example where the number of buyers $|B| = 4$ and the number of sellers $|S| = 8$. We would then expand these numbers to much larger sizes, keeping the other parameters fixed, to examine whether the size of marketplace influences the level of satisfaction of its agents. We may also want to see if a buying agent would be better off by exploring the market more often (i.e., using a higher value for ρ_{min}).

We expect that the results of the simulation will help us investigate the following interesting issues:

(i) Micro behaviours: We would like to know whether a buyer will achieve a better level of satisfaction when it uses a reputation mechanism, and whether a seller will have more chances to win an auction when it considers improving the quality of its products.

(ii) Macro behaviours: We would like to see how a market populated with our buyers and sellers will behave. In particular, we are interested in knowing if such a market would reach an equilibrium state (i.e., the agent population remaining stable), as some sellers who repeatedly fail to sell their products may decide to leave the market.

(iii) Performance: It is also interesting to evaluate how better a buyer can perform (in terms of computational time) if it uses a reputation mechanism.

To address these issues, we plan to compare the proposed algorithm with a simplified version where buyers do not use a reputation mechanism and sellers do not consider altering the quality of their products. For instance, to compare the satisfaction level of a buyer using the proposed algorithm with that of a buyer using the simplified version, we may run simulations to record and compare the true product values that each of these two buyers obtain after they each have made the same number of purchases in the same market. Alternatively, we may compare the probability that each of these two buyers is satisfied with the good it receives[3]. Similarly,

[3]A buyer is satisfied with the good it purchases if $\delta = v^b(p, q) - \vartheta^b(g) \geq 0$.

we may compare the satisfaction level of a seller using the proposed algorithm with that of a seller using the simplified version, by recording and comparing the actual profits that each of these two sellers make after they have participated in the same number of auctions in the same market. Performance evaluation can be done by measuring and comparing the average computational time needed to make a purchase decisions between two buyers, one using the proposed algorithm and the other using the simplified version.

We also plan to experimentally consider a number of additional versions of the algorithm in order to clearly specify the circumstances under which a particular version is preferable. The additional versions that are of interest to us include

- Buyers (sellers) do not keep track of sellers' (buyers') behaviour[4].
- Buyers keep track of sellers' behaviour but sellers do not keep track of buyers' behaviour.
- Buyers keep track of sellers' behaviour, while sellers divide buyers into groups and keep track of groups of buyers' behaviour.

Future Research Directions

For further research, it would also be possible to investigate more sophisticated algorithms for agents in electronic markets that allow agents to cooperate with other agents and/or take advantage of their knowledge about other agents to maximize their local utility. One specific case to consider is allowing buyers in the marketplace to form neighborhoods such that within a neighborhood they inform one another of their knowledge about sellers. These buyers can then use their own knowledge combined with the informed knowledge to make decisions about which sellers to select. We predict that this form of transferring knowledge will be especially beneficial to new buyers, who may be able to use the experience of existing buyers to make satisfactory decisions without having to undergo several trials to build up enough experience for themselves. Allowing agents to share knowledge with one another may necessitate the *social dimension* of current reputation models, i.e., the issue of how an agent should evaluate the reputation of another agent based on the ratings of the latter's neighbors. The work of Yu and Singh (Yu & Singh 2000) served as a motivation for our concept of reputation. It includes a mechanism for sharing information about reputation within neighborhoods and would therefore be a useful starting point for any future work which explores the use of advice from other buyers in the marketplace.

One additional avenue for future work is to explore further the concept of reputation in multi-agent electronic marketplaces. Modelling reputation in electronic marketplaces is important because it provides a basis for engendering trust between agents. We are interested in addressing questions such as *(i)* Is there a better way for buying agents to benefit from using a reputation mechanism? *(ii)* Can selling agents also make use of a reputation mechanism? *(iii)* What would be an efficient and suitable way to represent, manage, and use reputation in electronic marketplaces? This line of research may lead to an analysis of existing reputation models and the development of a new model. The reputation model of (Sung & Yuan 2001) is interesting, because it proposes that the degree to which an agent's reputation is adjusted should be dependent on the importance of the transaction. This suggests that it may be useful to allow the μ (cooperation factor) and ν (non-cooperation factor) to be variable rather than constants. Another useful starting point for further work on reputation is the research of Sabater and Sierra (Sabater & Sierra 2001), which proposes that reputation be modelled as a weighted combination of different factors.

[4] In our algorithm, a buyer (seller) keeps track of sellers' (buyers') behaviour by including variable s (b) in its expected value (expected profit) function.

Conclusion

In this paper, we proposed a feasible, reinforcement learning and reputation based algorithm for buying and selling agents in market environments. According to this algorithm, buying agents learn to optimize their expected product values by selecting appropriate sellers to do business with among their reputable sellers. Selling agents also learn to maximize their expected profits by both adjusting product prices and optionally altering the quality of their products. We discussed that the proposed algorithm may lead to improved performance for buying agents, higher level of satisfaction for both buying and selling agents, reduced communication load, and more robust systems. This work therefore demonstrates that reputation mechanisms can be used in combination with reinforcement learning techniques to design intelligent learning agents that participate in market environments.

Our future research aims to provide a set of feasible learning algorithms together with a clear characterization of different situations under which a particular algorithm is preferable. Such a characterization will address several important questions, such as under which circumstances buying agents should make use of a reputation mechanism, under what conditions agents may not need to track behaviour of other agents, and under what situations buying agents should exchange their knowledge about selling agents, etc. By accomplishing this objective, we hope to provide some general guidelines for AI-systems designers in building effective economic agents and desirable market environments.

This paper presents a strategy for modeling an economic marketplace populated by buyers and sellers. We have focused on how buying and selling agents can participate in auctions in order to purchase or sell goods. We have specified how reputation can be used as an important factor in the chain of exchange and how strategies for buying agents can be flexible to adjust to dynamic marketplaces where new sellers may arrive or current sellers may alter the quality of their goods. We have begun an analysis of an economic marketplace where agents employ this reputation-based reinforcement learning strategy. We are planning to conduct experiments to show the value of modeling reputation. The strategies presented in this paper also constitute useful computational tools for economic agents, with an aim to reach more effective overall decisions.

References

Chavez, A., and Maes, P. 1996. Kasbah: An agent marketplace for buying and selling goods. In *Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology*.

Davis, R., and Smith, R. 1983. Negotiation as a metaphor for distributed problem solving. *Artificial Intelligence* 20(1):63–109.

Doorenbos, R. B.; Etzioni, O.; and Weld, D. 1997. A scalable comparison-shopping agent for the world wide web. In *Proceedings of the First International Conference on Autonomous Agents*.

Kephart, J. O. 2000. Economic incentives for information agents. In *M. Klusch and L. Kerschberg, editors, Cooperative Information Agents IV, Lecture Notes in Artificial Intelligence, Vol. 1860*, 72–82. Springer-Verlag, Berlin.

Littman, M. L. 1994. Markov games as framework for multi-agent reinforcement learning. In *Proceedings of the Eleventh International Conference on Machine Learning*, 157–163.

Nagayuki, Y.; Ishii, S.; and Doya, K. 2000. Multi-agent reinforcement learning: An approach based on the other agent's internal model. In *Proceedings of the Fourth International Conference on Multi-Agent Systems*, 215–221.

Ono, N., and Fukumoto, K. 1996. Multi-agent reinforcement learning: A modular approach. In *Proceedings of the Second International Conference on Multi-Agent Systems*, 252–258.

Sabater, J., and Sierra, C. 2001. Regret: A reputation model for gregarious societies. In *Papers from the Fifth International Conference on Autonomous Agents Workshop on Deception, Fraud and Trust in Agent Societies*, 61–69.

Sandholm, T. W., and Crites, R. H. 1995. Multi-agent reinforcement in the iterated prisoner's dilemma. *Biosystems* 37:147–166.

Sen, S.; Sekaran, M.; and Hale, J. 1994. Learning to coordinate without sharing information. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, 426–431.

Smith, R. G. 1980. The contract net protocol: High level communication and control in a distributed problem solver. *IEEE Transactions on Computer* C-29(12):1104–1113.

Sung, H., and Yuan, S. T. 2001. A learning-enabled integrative trust model for e-markets. In *Papers from the Fifth International Conference on Autonomous Agents Workshop on Deception, Fraud and Trust in Agent Societies*, 81–96.

Vidal, J. M., and Durfee, E. H. 1996. The impact of nested agent models in an information economy. In *Proceedings of the Second International Conference on Multi-Agent Systems*, 377–384.

Weiss, G. 1993. Learning to coordinate actions in multi-agent systems. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, 311–316.

Wurman, P. R.; Wellman, M. P.; and Wash, W. E. 1998. The michigan internet auctionbot: A configurable auction server for humans and software agents. In *Proceedings of the Second International Conference on Autonomous Agents*, 301–308.

Yu, B., and Singh, M. P. 2000. A social mechanism of reputation management in electronic communities. In *M. Klusch and L. Kerschberg, editors, Cooperative Information Agents IV, Lecture Notes in Artificial Intelligence, Vol. 1860*, 154–165. Springer-Verlag, Berlin.