

Learning to be Competitive in the Market

From: AAI Technical Report WS-99-12. Compilation copyright © 1999, AAI (www.aaai.org). All rights reserved.

Eugénio Oliveira

José Manuel Fonseca

Nicholas R. Jennings

LIACC-University of Porto

New University of Lisbon

Queen Mary & Westfield College

Rua dos Bragas, Porto, Portugal

2825 Monte da Caparica, Portugal

Univ. of London, London E1, 4NS, UK

eco@fe.up.pt

jmf@uninova.pt

n.r.jennings@qmw.ac.uk

Abstract

Agents that buy and sell goods or services in an electronic market need to adapt to the environment's prevailing conditions if they are to be successful. Here we propose an on-line, continuous learning mechanism that is especially adapted for agents to learn how to behave when negotiating for resources (goods or services). Taking advantage of the specific characteristics of the price adaptation problem, where the different price states are ordered, we propose a specific reinforcement learning strategy that simultaneously allows good stability and fast convergence. Our method works by positively reinforcing all the lower value states if a particular state is successful and negatively reinforcing all the higher value states when a failure occurs. The resulting adaptive behaviour proved, in several different market situations, to perform better than non-adaptive agents and led to Nash equilibrium when faced with other adaptive opponents.

Introduction

In dynamic markets where agents can appear and disappear, where strategies can be continuously modified, and where external conditions can be unexpectedly changed, it is extremely difficult, sometimes impossible, to define *a priori* the best strategies to be used by buyer or seller agents. Consequently, agents need to be endowed with the ability to quickly adapt their behaviour in accordance with dynamic market changes. Such capabilities should also include responses to changes in the other agents' behaviour. Whether these reactions can be dealt with in the same manner as any other environmental factor or whether they should be dealt with in a special way, is an open issue. This controversy can be seen as a debate between model-based learning (to compute the agent's next action by taking into account a model of its acquaintances) and direct learning (to directly learn, from past experience, the expected utility of the agent's action in a given state) (Parkes and Ungar 1997).

Once the dynamics of the market-place are considered, the difficulties associated with learning increase still further. This is because an agent also has to determine how other agents in the market may also change their strategies through some learning capability. This possibility gives another dichotomy between what can be called "myopic learning agents" (Vidal and Durfee 1997) that use a simple,

short-term learning model and strategic learning agents that consider a long-term model including, in their own model, the learning process of other agents.

The work described in this paper follows the former approach and aims to endow agents in electronic markets with simple and effective learning capabilities. Such agents need to adapt to the changing conditions whether they are buyers or whether they are sellers. We believe this feature is essential if agents are to be successful. In our experiments, agents offering their own resources (time availability) in response to the announcement of future tasks to be executed, apply a reinforcement learning algorithm to determine the most appropriate price to bid under current market conditions (including the amount of possible future work). Therefore, we propose an on-line, continuous learning mechanism that is especially adapted for agents to learn how to behave when negotiating resources (goods or services). The aim of this work is then to give a contribution to enhance, through the agents' adaptivity, their performance in the electronic market environment.

In the following section, we discuss the learning process in multi-agent systems with a special focus on who, how and what can be learned in this kind of systems. In the third section, we present the learning algorithm and highlight the adaptations that were specifically required for this scenario. In the fourth section, a variety of illustrative experiments and respective results are presented and discussed. In particular, we sought to investigate the following issues: Does an adaptive agent, using an algorithm that includes exploratory stages, perform well when competing with other fixed strategy agents in the electronic market environment? Is this true independently of what is really changing in the market? (either the other agents' behaviour or the external market conditions). Can we still verify the theory concerning the tendency to reach the Nash equilibrium when all the seller agents use the very same selling strategy? The fifth section reviews the research background and related work. Finally we present our conclusions and pinpoint avenues of future work.

How, who and what to learn

We view agent-mediated electronic markets as a particular kind of multi-agent system. Due to the dynamics of the market, where different agents (both in number and in nature) may offer or ask for services (goods, resources) that are more or less scarce, agents should adapt their bids

to the current situation in order to be more competitive. In the context of a multi-agent system, we may distinguish whether the learning process:

- i) should be performed by an agent (or each one of them) in order to improve its own utility, or
- ii) should be performed through the co-operative activity of several agents in order to increase the overall utility function.

The former case can be further sub-divided into those situations where an agent learns with its own experience and those where it learns taking into account the actions (or both actions and states) of the other agents playing in the same scenario. In either case, this is still an isolated agent learning capability. As an example of learning in a multi-agent context, (Prasad and Lesser 1999) present the co-operative selection of the most appropriate pre-existing co-ordination strategy. However, more than just selecting from among pre-existing strategies, learning in multi-agent environments should imply that agents collaborate in learning something they are not able to learn alone. This capability still encompasses two different possibilities: either agents have a limited scope about the environment (it could be the search space or the other agents' actions) or they have different and uncertain perspectives about the entire environment (either the search space or all the available information on the other agents' actions). In either case, however, the problem can be formulated as: how to design a procedure that facilitates a kind of meta-learning that makes it possible to coherently integrate partial perspectives into common, generally available knowledge, that lead the agents to the satisfaction of an overall goal. The question now becomes: can we identify a joint goal and, consequently, the need for joint learning in an Electronic Commerce scenario? This is not what usually happens with buyers and sellers. However, we may look at the market itself as having a global goal: to promote as many deals as possible (perhaps because a small fee is collected for each accomplished deal). Therefore, all the multi-agent system – here the agents selling/buying in the market and the market-place (another agent) itself (!) – may learn from all the past agents' joint actions. These joint actions are in fact also reflected in the existence (or not), as well as in the specific conditions, of the final deals.

Moreover, procedures for selecting agents' interaction protocols could also be learned and made available at the multi-agent system level. Focusing now on what a buyer agent can learn, we can say that the buyer agent can learn about the market-place in which it operates in order to better select which auction protocol to use and about which negotiation partners (sellers) to negotiate with. If an agent has little experience about the market, it has to specify the product or service it is looking for with a high degree of tolerance in order to maximise the number of proposals it receives. Whenever a large number of proposals is expected, a multiple round negotiation protocol (such as the English auction) is more appropriate than a single offer protocol because this way the buyer can stimulate the negotiation between the different bidders and obtain better

proposals than when a single offer protocol is used. However, if the agents are well informed about the market, they can use a single-offer protocol because they have the knowledge to describe and send out a more detailed product specification. This will reduce the time spent in communication. Moreover, if the buyer already has adequate knowledge about the most promising sellers, it can restrict its announcement to those specific agents (instead of broadcasting it to the entire community). Thus, it can be seen that learning can save both communication overhead and time. However, trying to learn too much can also be dangerous. If a buyer agent trusts its current acquired knowledge too much, it may disregard market evolution and miss important changing market characteristics. In order to avoid this problem, the learning algorithm must provide ways to explore the market and test less promising solutions in order to verify whether they have become adequate. Failure to explore new possibilities can lead to important losses but, on the other hand, exploration also involves potential losses, therefore it must be employed carefully.

The choice between the different negotiation protocols can also be influenced by strategic factors if the buyer agent has good knowledge about the market conditions. For example, when a change occurs in the market and new agents appear, it is expected that direct competition between those agents will provide some advantages to the buyer. Therefore, in such situations, iterated protocols are preferable to single-offer auctions. In situations where agents that are usually very competitive quit or become inactive, single offer protocols can take advantage of less aware sellers that keep trying to beat non-existent rivals. It is of course also true, that this type of strategic behaviour implies risks that must always be taken into account.

Learning can also be useful on the vendor's side because it can enable these agents to continuously improve their competitiveness. Depending on the protocol adopted by the buyer, a seller can take different advantages from a good knowledge about the market in which it is competing. Here we focus our attention on two possible alternative negotiation protocols: the English and the First-price sealed bid auctions (Sandholm 1996). If an English auction is adopted, the seller can improve its negotiation efficiency by reducing the number of offers. This saves both on processing time and communication overhead. Bids with very low probability of being accepted will be avoided by using *a priori* knowledge about the market. If the buyer adopts a First-price sealed bid auction, the single offer that sellers are allowed to make is extremely important since it cannot subsequently be changed. Therefore, the most appropriate value has to be calculated based only on the vendor's knowledge and constraints. The ideal value for this bid would be one that is slightly better than all other bids in order to beat them. As it is the case for the buyer, the vendor's knowledge must also be continuously updated in order to keep track of the changes in the market. Learning about the current market conditions can make a significant impact on the achievement of better deals.

We have so far identified some of the issues that can be learned in the multi-agent based Electronic Market framework. The problem now becomes how to make agents learn. In agent-based electronic markets, learning is usually based on the results of each agent's previous experiences. As already discussed, the learning process must be continuous in order to allow the agents to adapt themselves to a possibly evolving market. Thus, the sporadic testing of different possibilities, even when they do not appear to be the most promising action, should be encouraged by the decision making process. Such exploration is the only way of trying out new possible and non-obvious solutions, evaluate them and find out either new potential attitudes or have poorly scored experiences confirmed. However, this exploration needs to be well balanced with exploitation to ensure the potential losses introduced by the former are not unacceptably heavy.

In the context of this work, we chose reinforcement learning (RL) (Sutton and Barto 1998) techniques to allow agents to adapt to dynamic market conditions. Although different learning algorithms such as C4.5 (Quilan 1993), CART (Breiman et al. 1984), Case Based Reasoning (Kolodner et al. 1985) and many others could be applied to this problem, we have chosen RL because it is naturally a continuous and incremental process where exploration is a native concept. These characteristics mean RL is well suited to dynamic environments. However, unlike dynamic programming methods, RL does not require a model of the environment's dynamics, nor the next state probability distribution function.

In this section we have proposed several different goals for agent learning in the context of a multi-agent based market. However, in our work, we focus specifically on the seller's bidding policy adaptation to an unknown market where buyers do not reveal any information about their opponents' offers or the market demands. The sellers must infer the correct behaviour based only on their success or failure in successive negotiation episodes. As it is the case in real markets, the simulated market is dynamic; unpredictable changes occur both in the market demand and in the opponents' strategies. This scenario was chosen because realistic Electronic Markets will be open and the agents' performance will highly depend on their capability to respond to those dynamic changes. To demonstrate the applicability of the reinforcement learning method in this scenario, a single criterion negotiation scenario, where just the price of a good is negotiated between one buyer and a group of selling agents, was simulated. A first-price sealed bid auction protocol was adopted for testing the algorithms because this is the most demanding situation in terms of knowledge required about the market. The challenge for the buyer agents is, therefore, to learn the most appropriated price to bid in an unknown, dynamic market.

The reinforcement learning algorithm

Reinforcement learning is based on rewarding actions that turn out to be positive and punishing those that are negative. To deal with the learning problems described

above, classical RL techniques were adopted. However, some particular characteristics of the Electronic Market scenarios were exploited in order to improve efficiency. When using RL, the value associated to the outcome of each possible action is a key measure. It is usually calculated by the following formula (Sutton and Barto 1998):

$$Q_k(a) = \frac{r_1 + r_2 + \dots + r_{k_a}}{k_a} \quad \text{where} \quad \begin{cases} Q_i(a) - \text{estimated value for action } a \\ r_n - \text{reward received on trial } n \\ k_a - \text{number of trials for action } a \end{cases}$$

This formula can be rewritten in order to eliminate the need to store all the results from the beginning of the learning process:

$$Q_{k_a}(a) = Q_{k_a-1}(a) + \frac{1}{k_a} [r_{k_a} - Q_{k_a-1}(a)]$$

The coefficient $1/k_a$ determines the decreasing influence of the estimation error as experience grows. Although in static scenarios this is a desirable characteristic, in dynamic situations the past experience must be partially forgotten in order to correctly consider new experiences. This is achieved by adapting the equation in the following manner:

$$Q_{k_a}(a) = Q_{k_a-1}(a) + \alpha [r_{k_a+1} - Q_{k_a-1}(a)] \quad \text{where } 0 \leq \alpha \leq 1$$

It can be proved that Q_k converges to its true value if and only if:

$$\sum_{k=1}^{\infty} \alpha_k(a) = \infty \quad \text{and} \quad \sum_{k=1}^{\infty} \alpha_k^2(a) < \infty$$

Notice that, if α is constant, the second equation is not satisfied. However, this is not a problem in our case because we are dealing with a dynamic scenario where the true value of the actions can be continuously changing. It is like a kind of a moving target that must be followed.

The choice of which action to adopt in each situation is another important problem in RL. Suppose that several alternatives can be adopted in a particular situation: greedy behaviour can be achieved by always choosing the most promising one. This is a very simple behaviour that does not allow any exploration to be performed. However, too much exploration is a disadvantage because it means the learner may make too many poor choices. To this end, the *Softmax* action selection policy is commonly adopted:

$$p_t(a) = \frac{e^{Q_t(a)/\tau}}{\sum_{b=1}^n e^{Q_t(b)/\tau}} \quad \text{where } \tau \geq 0$$

$p_t(a)$ is the probability of choosing action a and τ is a positive parameter usually called temperature. When a high temperature value is used, all actions are chosen with approximately equal probability. With low temperature values, more highly evaluated actions are favoured and a zero temperature corresponds to greedy behaviour.

The stated goal of our experiments is to study the agents' price value adaptation to market conditions. The problem was modelled by defining N possible actions a_i with $1 \leq i \leq N$ with associated prices $P_i(a_i)$, from the minimum acceptable price to the maximum possible price. The goal of the learning process is to choose the most

adequate action (and consequently the most adequate price) to bid in each situation. The initial value for each action was set to the expected profit from selling at that price $P(a_i)$. This means that the higher the price, the higher the initial expected Q value for the corresponding action. It was also decided to initialise the agents' in the state corresponding to their average price and let them explore the market.

In order to optimise the agents' economic performance in our scenario, the learning algorithm has been slightly adapted. In fact, when an agent makes a deal at a particular price, there is no point in exploring lower prices or in trying much higher prices. Thus the exploration possibility is limited to the next stage (higher price) and only when the agent receives a positive reward. Therefore, instead of exploring all possible states with some probability (calculated with *Softmax* or any other formula) we decided to explore only the next state, with a pre-defined probability, whenever an agent receives a positive reinforcement. The usual state transitions, performed by exploitation, were also limited to two situations: a) when the agent receives a negative reward and the previous state has higher quality than the current one or b) when the agent receives a positive reward and the next state has a higher expected quality. Again, this is due to the fact that if the agent is making deals at a specific price, there is no point in reducing it. In a similar way, if an agent is failing to get deals at its current price, there is no reason to raise the price even if the corresponding state seems more promising.

In order to obtain a faster convergence of the learning algorithm and produce more stable agent behaviour, some improvements have been introduced in the reward/penalty policy. If an agent does not make a deal at state a_c (in our scenario, it loses one or more announced tasks for a given period) because it is bidding too high, a penalty is given to the current state and to all the states corresponding to higher prices. The idea is that if the current price is unacceptably high, all the higher prices will also be unacceptable. Because penalties are calculated based on the loss L caused by unemployment, the same penalty value is imposed to all the penalised states. In a similar fashion, if a deal is obtained at state a_c the corresponding positive reward is given to that state and to all the states corresponding to lower prices. The idea is now that, if a customer accepts a price it would also accept any lower price. In this situation, the profit $P(a_i)$ obtained by a deal at the price corresponding to each state a_i is used as the value for the positive reinforcement.

The learning algorithm was designed the following way:

- sellers bid according to the price defined for their current state $P_i(a_c)$.
- when a seller gets a task at state a_c this state and all the lower price states receive a positive reinforcement:

$$\forall_{i \leq c} \quad Q_k(a_i) = Q_{k-1}(a_i) + \alpha [P(a_i) - Q_{k-1}(a_i)]$$

If the next state (next higher price) has a higher expected value ($Q(a_{i+1}) > Q(a_i)$) it is immediately adopted.

- Otherwise exploration is performed with a probability σ .
- when an unemployment period is reached and there were task announcements for that period, the current state a_c and all the higher price states receive a negative reinforcement corresponding to the losses suffered:

$$\forall_{i \geq c} \quad Q_k(a_i) = Q_{k-1}(a_i) + \alpha [L - Q_{k-1}(a_i)]$$

If the previous stage promises more than the current one ($Q(a_{i+1}) > Q(a_i)$) it is immediately adopted. Otherwise the current state is maintained.

In our case, both α and σ were set as system parameters for each adaptive agent. As we will see later, both parameters determine the stability of the agents' behaviour. If a higher learning rate α is used, the agents have a faster reaction to market changes but they are unstable because they tend to over react to sporadic, but insignificant, episodes. A low learning rate makes the agents too slow in changing their beliefs. A high exploration probability σ determines that the agents make numerous testing bids, leading to significant losses (especially in stable markets). However, a high exploration probability gives the agents the ability to react faster to market changes because they lose less time in detecting new situations. Therefore, a good compromise, usually depending on the scenario, must be found for each situation in order to obtain the best performance for the adaptive agents.

Simulation results

To demonstrate that an agent endowed with learning capabilities can successfully adapt itself to an unknown, competitive market, exhibiting an optimal (or nearly optimal) behaviour, several simulations were made. In these simulations, different agents (sellers) try to sell their services (available time) to different buyers (announcing tasks to be executed at a particular time in the future) using a first-price sealed bid auction. Sellers can only serve one customer at a time. Therefore, they must sell their services at the highest possible price, while trying to avoid periods of unemployment. To do this, seller agents must distinguish between an "adverse defeat" and a "favourable defeat". The former occurs when an agent loses an announced task and has no other alternative for the same period. The latter occurs when an agent loses an announced task but subsequently wins an alternative one. Our simulation scenario is inspired by a real-world application in which several physical resources such as excavators, trucks, workers, etc, compete to win tasks in an open electronic market (Oliveira, Fonseca, and Garção 1997).

The first experiment compares two different types of selling agents: a fixed-price agent and an adaptive one. The fixed-price agent offers its services for a constant price of 2200 units. The adaptive agent is configured with a large negotiation margin between a minimum price of 350 and a maximum price of 3200. Its initial price is set at an average value of 1700. The adaptive agent tries to learn, online during the session, how to obtain the maximum profits (using $\alpha=0.15$ and $\sigma=0.1$). In this scenario, the amount of

work the buyer agent announces in the market is sufficient to guarantee approximately 90% occupancy to both sellers.

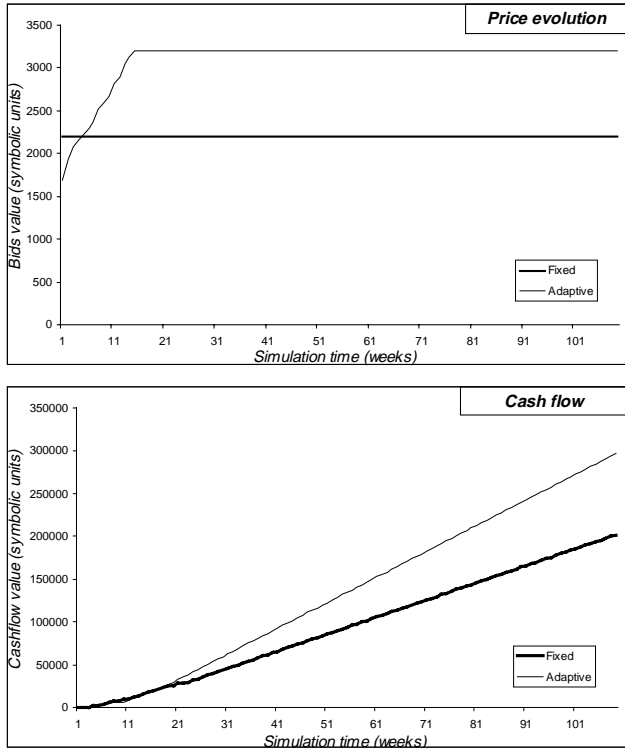


Figure 1 – Adaptive versus fixed-price agents in a high occupancy scenario.

Figure 1 shows that the learning agent performs considerably better than the fixed-price agent (compare the cash flows). As there is a high amount of work launched on the system, the adaptive agent realises, through its learning capability, that there is no need to be competitive and fight with the other agent. Therefore, it quickly raises its price to its maximum value (3200). Notice that for a short period at the beginning of the simulation, the adaptive agent obtains worse results. This is a consequence of its initial adaptation period when it is using prices that are too low.

The second experiment examined the case where a lower amount of work (smaller number of tasks, guaranteeing around 40% occupancy for each agent) was launched on the system. The other scenario characteristics are as before. Figure 2 shows that after a short period of uncertainty, caused by the delay between the task announcements and the periods of unemployment, the adaptive agent realises that the best solution is to set a price just lower than its opponent. Notice that the adaptive agent has no knowledge either about how many competitors are in the market or what their strategies are. Once again, during the learning period the adaptive agent performs worse than the fixed price agent. However this fact is compensated in the long term. The periods where the adaptive agent sets higher prices than the fixed-price agent (see the oscillations in the upper graphic of figure 2.) are due to exploration and market instability. Exploration because, from time to time, the adaptive agent bids with a higher price in order to test

the market. Market instability because the volume of work launched in the system has a stochastic time distribution and sporadic concentrations of announcements induce incorrect conclusions in the adaptive agent. Again, in this experiment, the conclusion is clearly drawn from the cash flow results: the adaptive agent performed best.

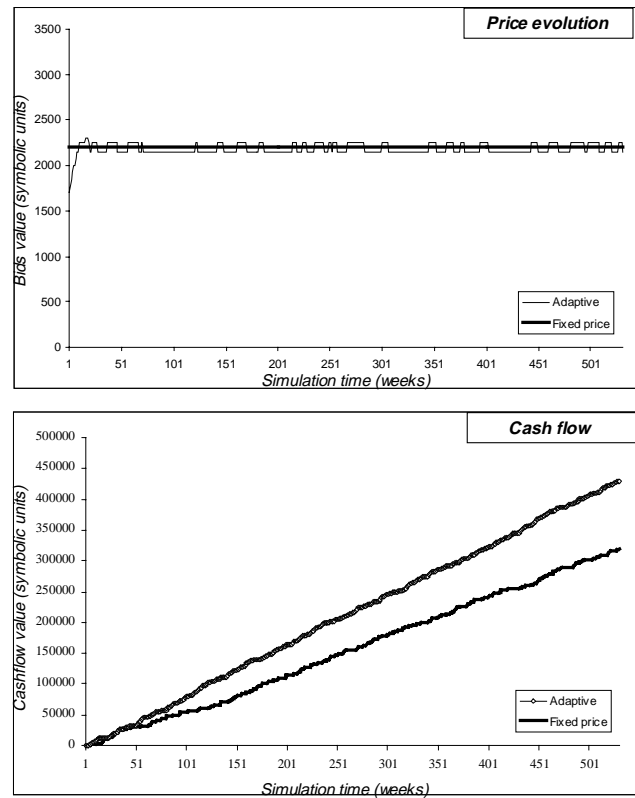


Figure 2 - Adaptive versus fixed-price agents in a low occupancy scenario.

In the third experiment, the adaptive agent, using different α values, is confronted with a “step changing price agent” (SCPA) in order to study the influence of this parameter on its adaptive behaviour. Figure 3 shows the performance of the same agent when using different alpha values ($\alpha=0.05, 0.1, 0.25, 0.5$). As predicted, a lower value for α leads to a slower reaction to market changes and a higher value for α leads to a faster reaction. However, a fast reaction does not automatically mean better results. Actually, the results of the different adaptive agents are broadly similar but the agent using $\alpha=0.25$ obtains the best cash flow results. Initially, and after each SCPA change, the $\alpha=0.5$ agent obtains the best results (highest slope on the cashflow curve) because it is the fastest to propose the ideal price. However in the long run, its over reaction in the face of sporadic positive or negative results leads to losses that are caused by inappropriate bids.

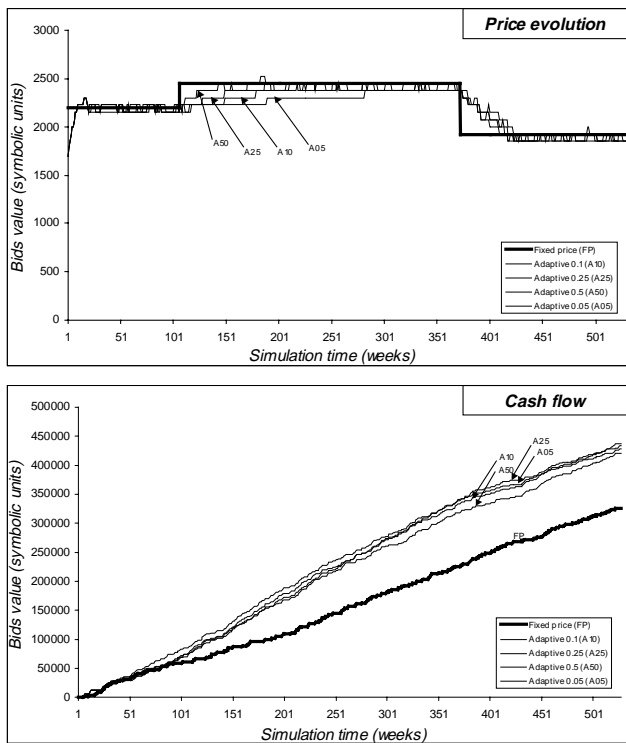


Figure 3 – Comparison between different adaptation coefficients.

In our fourth example (Figure 4), the dynamics of the market come from the varying number of buying agents (or task announcements). We now compare the reactions of two different agents (one using an adaptive strategy and the other using a fixed-price strategy) when the amount of available (announced) work is dynamically changed (not displayed in the figure). In the first weeks (approximately until week 150) the amount of work is low and the adaptive agent keeps its price slightly under its opponent's price. It occasionally explores higher prices. After week 150, the amount of work is raised (introducing new buyers in the market) and the adaptive agent realises that it can raise its price up to its maximum. At week 670 the extra buyers are removed from the market, re-establishing the initial scenario conditions. Consequently, the adaptive agent lowers its price until it achieves a value lower than its opponent. Looking at the cash-flow chart, we can see that the adaptive agent always gets better results than its fixed-price counterpart. It steadily increases the gap between their respective cash-flows, except around weeks 670 to 750 (period A in the figure) where the cashflow curve slope of the fixed-price agent is slightly higher than that of the adaptive agent. This is caused by the adaptive agent asking a too high price until it adapts itself to the new situation (what happens around week 750).

Our final simulation considers a market exclusively composed of four adaptive agents with similar price bidding and learning characteristics (figure 5). This simulation was carried out in order to test the convergence of the proposed learning algorithm as well as the validity

of the agent's behaviour when compared with predictions coming from economic theory.

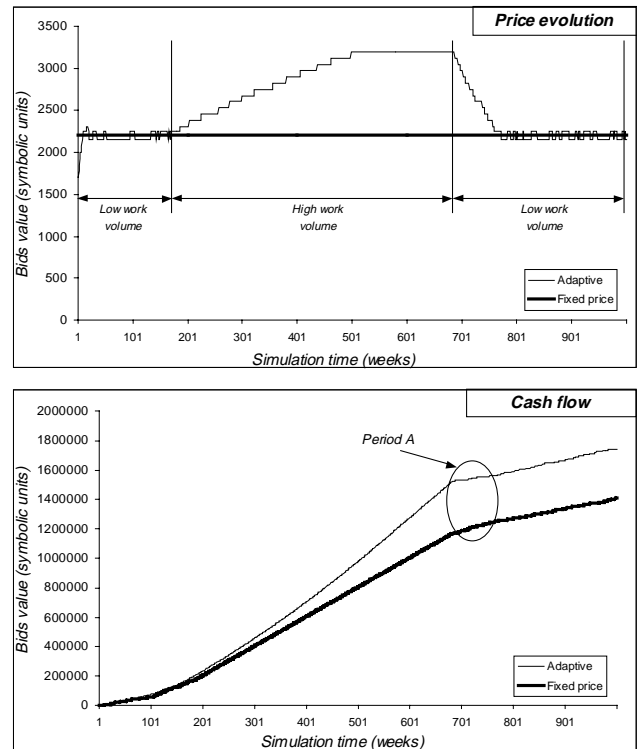


Figure 4 – Influence of the market changes on the agent's economic behaviour.

When a market is composed of a limited number of competitive agents, where price is the only competitive factor, it is called an oligopolistic market (Pindyck and Rubinfeld 1995). In such markets, if the agents have similar price characteristics, it is expected that agents reach the Nash equilibrium at their reservation price. Indeed this is exactly what happens in our experimental market (see figure 5). Because the agents do not get any profits when they are unemployed, and just a small profit is obtained when selling near the reservation price, the cashflows of all the agents tend to the horizontal. This "collective suicide" tendency is easily explained. Suppose that an agent is bidding with exactly the same price as all its competitors. In this case, it will obtain a share of the market approximately equal to all the others. However, if it lowers its price a small fraction, it can obtain a significant extra share that more than compensates for the reduction on the per unit profits. Unfortunately, all the other agents will follow the same policy and their shares will become equal again, but now at a lower price value for everyone. This process continues until all agents reach their reservation price (350 in this experiment).

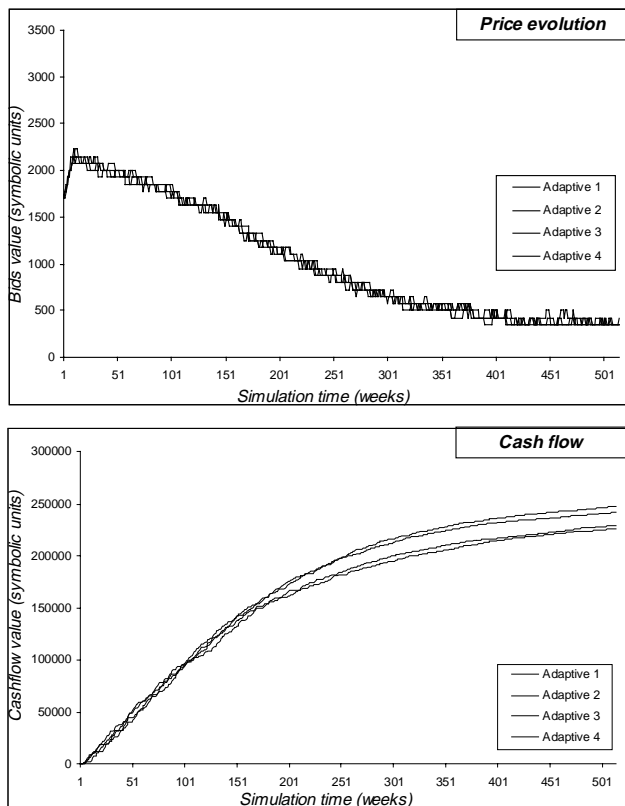


Figure 5 – An oligopolistic market with four adaptive agents.

Related work

Learning in multiple agent systems is a relatively new field of endeavour that has already achieved significant importance both in the distributed artificial intelligence and the machine learning communities (Weiss 1997). This is so because, since complete knowledge about the preferences and rationality of other agents is rarely available, learning can be used as a way for the agents to adapt themselves to new and unpredictable situations. Following Claus and Boutilier (1998), learners can be called independent when learning from tuples $\langle a_i, r \rangle$ (where a_i is the action of the agent i and r is the following reward) and joint learners when learning from tuples $\langle a, r \rangle$ (where a is the joint actions of all the agents). This is equivalent to saying that isolated learning is performed by an agent taking into account its own environment which can include other agents that act – and even learn – in that environment. Moreover, a number of papers consider exactly what should be learnt about other agents. Durfee and Vidal (1997) introduce the notion (or the terminology) of K -level agents. A 0-level agent, although being adaptive, does not model the underlying behaviour of other agents. Rather it selects its next action by using a function of the other agents' actions (a_{-i}) (not states). In fact, 0-level just estimate other agents' actions at time t , a_{-i}^t , as a linear

function of previous observations: A 1-level agent models the other agents' actions in a deeper way (a^{-i}). It attempts to learn the relationship between an agent's action at time t and its corresponding state s_t^i , $a_t^i = f^i(s_t^i, a_{-i}^i)$. Of course, 1-level agents consider that the other agents have a stationary function f^i . We can understand by means of recursion what a k -level agent is for any k . Nevertheless, several simulations lead to the conclusion (Hu and Wellman 1998) that, under uncertainty (even if it is as little as 5%) about the other agents' behaviour, the best policy for a learning agent is to stay at the 0-level. Thus agents should avoid sophisticated reasoning about their acquaintances. Nevertheless it has been shown that although learning agents with minimal assumptions about the others tend to perform better than agents that associate too much sophistication to their counterparts, when such attributes are warranted, agents can do better by learning more sophisticated models (Hu and Wellman 1998). Alternatively, Carmel and Markovitch (1998) propose a "model based learning agent" that instead of holding a model, maintains a distribution over a set of models that reflects its uncertainty about its opponent's strategy. It then uses reinforcement learning to update its knowledge. Instead of using sophisticated strategic-learning agents, we have also opted for the myopic-learning type of agents that use a simple, short-term learning model.

The growing importance of Electronic Commerce and negotiating agents has led to many different negotiation strategies (Rosenschein and Zlotkin 1994; Kraus 1997; Matos, Sierra and Jennings 1998), to argumentation-based negotiation (Sierra et al. 1998). In such scenarios, where intelligent agents represent buyers and sellers that try to maximise their own benefits in a completely selfish way, no co-operation is expected and no global optimisation is usually achievable.

Having learnt the lessons from the previous authors' work, we have here advocated, for the Electronic Market environment, the use of an online incremental learning algorithm for 0-level type of Agents. Our slightly adapted Q-learning algorithm showed to be adequate for selling agents' automatic bidding adaptation in a market with a varying number of competitors with unknown behaviours.

Conclusions and future work

This paper has discussed a number of different opportunities for buyers and sellers to automatically adapt their negotiation behaviours in dynamic, competitive markets in order to maximise their profits. Moreover, we have proposed what we believe to be an appropriate learning strategy (reinforcement Q-learning) for such application domains. Taking advantage of the ordering characteristics of the price adaptation problem, we proposed a specific reinforcement learning strategy that simultaneously allows good stability and fast convergence. The resulting adaptive behaviour proved, in several

different dynamic market situations, to perform better than their competitors and it led to Nash equilibrium when faced similar opponents.

As future work, we aim to study the influence of the different (Q-Learning algorithm) parameters on the agent's behaviour. We also want to examine adaptive parameter setting in order to speed up the agent's reactions without compromising their efficiency. The comparison of our learning algorithm with different adaptation strategies is another interesting line of research that we intend to pursue. At this time, we have shown that our adaptive agents are efficient against agents with static or nearly static policies, as well as against agents that adapt in a similar fashion. However, our adaptation strategy should also be compared with other strategies that may be more efficient in some (or even in all) situations. Finally we would like to explore the full potential of the Q-learning algorithm by updating the Q values several times during each episode. To do so, we have to determine whether it is feasible to have feedback (to compute rewards) from some intermediate negotiation steps and update our estimates accordingly.

References

- Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J., "Classification and Regression Trees". CA: Wadsworth, 1984.
- Carmel, D. and S. Markovitch (1998). "How to Explore Your Opponent's Strategy (almost) Optimally". Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS'98), Paris, 1998, pp. 64-71.
- Claus C., Boutilier C., "The Dynamics of Reinforcement Learning in Cooperative Multiagent Systems", Proceedings of the Fifteen International Conference on Artificial Intelligence (AAAI'98), pp 746-752.
- Hu, J. and Wellman, M. P., "Online Learning about Other Agents in a Dynamic Multiagent System". Proceedings of the Second International Conference on Autonomous Agents (Agents'98), Minneapolis/St. Paul, 1998, pp. 239-246.
- Kraus, S., "Negotiation and Cooperation in multi-agent environments", Artificial Intelligence Journal, Special Issue on Economical Principles of Multi-Agent Systems, Vol. 94 (1-2), pp. 79-98, 1997.
- Kolodner J. L., Simpson R. L. and Sycara-Cyranski, "A Process Model of Case-Based Reasoning in Problem Solving", Proceedings of IJCAI-85, Los Angeles, CA, 1985, pp. 284-290.
- Matos, N., Sierra, C., and Jennings, N., "Determining Successful Negotiation Strategies: An Evolutionary Approach". Proceedings of the Third International Conference on Multi-Agent Systems ICMAS'98, Paris, 1998, pp. 182-189.
- Oliveira, E. d., Fonseca, J. M., and Garção, A. S., "MACIV - A DAI Based Resource Management System", International Journal on Applied Artificial Intelligence, vol. 11, pp. 525-550, 1997.
- Parkes D., Ungar L., "Learning and adaptation in Multiagent Systems", Proceedings of the AAAI'97 Workshop on Multi-agent Learning, Providence, Rhode Island, 1997.
- Pindyck, R. and D. L. Rubinfeld, "Microeconomics", New Jersey: Prentice Hall, 1995.
- Prasad, M.N., Lesser V., "Learning situation-specific coordination in cooperative Multi-agent Systems" in Autonomous Agents and Multi-Agent Systems (to appear).
- Quinlan, J. R., "C4.5: Programs for Machine Learning". San Mateo: Morgan Kaufman, 1993.
- Rosenschein, J. S. and Zlotkin, G., "Rules of Encounter" CA: MIT Press, 1994.
- Sandholm, T. W., "Negotiation Among Self-Interested Computationally Limited Agents". PhD Thesis: University of Massachusetts at Amherst, 1996.
- Sierra, C., Jennings, N., Noriega, P., and Parsons, S., "A framework for argumentation-based negotiation", Intelligent Agents IV, LNAI Vol. 1365, pp. 177-192, 1998.
- Sutton, R. S. and A. G. Barto (1998). "Reinforcement Learning: An Introduction", MIT Press, Cambridge, MA, 1998.
- Vidal J., Durfee E.. "Agents learning about agents: a Framework and Analysis", in Proceedings of the AAAI'97 Workshop on Multi-agent Learning, Providence, Rhode Island, 1997.
- Vulkan, N. and Jennings, N. R., "Efficient Mechanisms for the Supply of Services in Multi-Agent Environments," Proceedings of the 1st Int. Conf. on Information and Computation Economies, Charleston, South Carolina, 1998, pp. 1-10.
- Weiss, G., "Distributed Artificial Intelligence Meets Machine Learning", LNAI, Vol. 1221. Berlin: Springer-Verlag, 1997.