Agents advanced features for negotiation in Electronic Commerce and Virtual Organisations formation process

Eugénio Oliveira, Ana Paula Rocha
LIACC, Faculty of Engineering, University of Porto

Abstract

Electronic Commerce technology has changed the way traditional business is being done. Transactions' complexity is increased due both to the huge amount of available information and also to the environment dynamics. Moreover, Electronic Commerce has enabled the arising of new economical structures, as it is the case of Virtual Organisations.
Our research aims at providing flexible and general-purpose systems for intelligent negotiation, both for Electronic Commerce and Virtual Organisation formation.
This paper proposes an Electronic Market architecture implemented through a Multi-Agent system. This architecture includes both a specific market agent that plays the role of market coordinator, as well as agents representing the individual business partners with their own goals and strategies. We also include a sophisticated negotiation protocol through multi-criteria and distributed constraint formalisms. An online, continuous reinforcement learning algorithm has been designed to enable agents to adapt themselves according to the changing environment, including the competitor agents.

1 INTRODUCTION

Electronic Commerce was often identified, in the recent past, with the capabilities of making available and/or searching for product information in electronic format through inter and intra nets.
Last years research efforts gave birth to new computer frameworks and platforms making it possible for electronic agents, acting as individual or business organisations delegates, to automatically negotiate through the Internet. However, most of the current available systems still are inflexible from several perspectives: language, strategic behaviour, adaptability, specific purpose. Electronic commerce techniques have to be seen as enabling technologies for more advanced inter-organisational relationships leading to the effective creation of virtual organisations.
A Virtual Organisation (VO) is an aggregation of autonomous and independent organisations connected through a network (possibly a public network like Internet) and brought together to deliver a product or service in response to a customer need. Virtual Organisation management should be supported by efficient information and communication technology. This aggregation of organisations is advantageous in the sense that it will reduce complexity – today's products and services are increasingly complex and requires close coordination across many different disciplines – and most important, will enable the response to rapidly changing requirements. The Virtual Organisation will only exist for a temporary time limit, that is the time needed to satisfy its purpose.
The VO life cycle is decomposed in four phases (Faisst, 1997) (Fischer, 1996), namely:

- Identification of Needs: Description of the product or service to be delivered by the VO, which guides the conceptual design of the VO.
- Partners Selection: Rational selection of the individual organisations (partners) which will compose the VO, based in its specific knowledge, skills, resources, costs and availability.
- Operation: Control and monitoring of the partners' processes, including resolution of conflicts, and possible VO reconfiguration due to partial failures.
- Dissolution: Breaking up the VO, distribution of the obtained profits and storage of relevant information.

Our work is currently focused on the second above mentioned phase (partners selection), and our proposal concerns the development of a Multi-Agent System (MAS) architecture to model the Electronic Market that supports the automation of the VO.
In the process of virtual organisation formation, goals representing the future virtual organisation purpose have to be described along several issues and made available to all those single organisations (enterprises) that are connected to the market. In order to compete, trying to bring their own specific competencies to the future consortium, agents representing organisations are endowed with several different possible behaviour tactics and higher level strategies for

negotiating in a multi-dimensional space. In fact, price is no longer the solely characteristic of importance for the consortium.
Moreover, those negotiable issues may not be independent and mutual constraints are often attached to more than one issue the potential partners have to agree upon. Agents representing those partners (enterprises, customers, suppliers, buyers and sellers) have to be able to gradually approximate their own proposals taking into account a complex graph of inter-relationships between different variables.
Another aspect we are paying attention to is the agents' capability to automatically adapt to the market's dynamics. Static behaviours, although ingenious, can easily be beaten either by other agents constantly adapting their own behaviour, or by the rapid change of other environment factors.
An online, continuous reinforcement learning algorithm (an adaptation of Q-learning) is being tested in a number of scenarios aiming at proving that adaptive agents may easily become winners when competing with agents using fixed behaviours.
To apply these learning methods as well as multi-criteria and constraint based policies to agents representing either organisations or individuals searching for an agreement in a future action is what we intend to deliver through our multi-agent based system suitable for electronic commerce and virtual organisation formation.
In the following section, we will introduce our proposed MAS architecture to model the Electronic Market, focusing our description in the agent's structure and behaviour. A brief survey on agent-based architectures for electronic commerce it is for example given by Guttman (Guttman, 1998). Then we describe a negotiation protocol that includes both multi-criteria and constraint-based formalisms. We also detail the negotiation decision' making, in particular the multi-criteria evaluation function and the reinforcement learning algorithm (Q-learning) for which we include some experimental results from preliminary related work. We finalise the paper with some conclusions and directions for future work.

2 ELECTRONIC MARKET ARCHITECTURE

Economic organisations are forced to respond to increasingly more complex and rapidly changing requirements. Virtual Organisations can satisfy these new challenges, as they combine the core competencies of several autonomous and heterogeneous organisations aggregated in a temporary network, presenting high flexibility and agility.
An Electronic Market infrastructure can efficiently support VO formation and operation. An Electronic Market (Bakos, 1998) can be seen as the virtual marketplace where business participants that are geographically distributed and possibly not known in advance can meet each other and cooperate in order to achieve a common business goal. An electronic network is used as an information and communication infrastructure, in order to link these distributed entities. In our proposed Electronic Market, autonomous organisations are the participants and the common business goal is the formation of a VO as the most favourable group of organisations that can satisfy a specific need. Besides individual organisations, other participants operate in the Electronic Market, whose role is to facilitate a set of auxiliary market functionalities like searching for goods or participants, filtering information or helping in negotiation.
Agents are autonomous or semi-autonomous entities that can perform tasks in complex and dynamically changing environments. Agent technology seems to be an appropriate paradigm for use in modelling individual organisations that participate in the Electronic Market, since they exhibit some relevant capabilities like autonomy, adaptability and ability to interact with others.
A Multi Agent System (MAS) – see (O'Hare 1996) for a general survey – consists of a group of agents that combine their specific competencies and cooperate in order to achieve a common goal. Efficient cooperation as well as coordination procedures between agents endows a MAS with a capability higher than the sum of the individual agent's capabilities.
We therefore propose a Multi-Agent System architecture to model the Electronic Market that supports the establishment of the VO (see Figure 1). This MAS includes several heterogeneous and semi-autonomous intelligent agents representing the different independent organisations, and also a special agent called **Market Agent** that helps in the VO formation process. Each one of these organisations are then represented by means of two kinds of agents: the organisation itself which is responsible for the "Identification of Needs" phase (which will not be dealt with here in this paper), and the organisation representative agent for that specific previously identified need (which is called **Organisation Agent**).
We will now describe in more detail the structure and functionality of these agents.

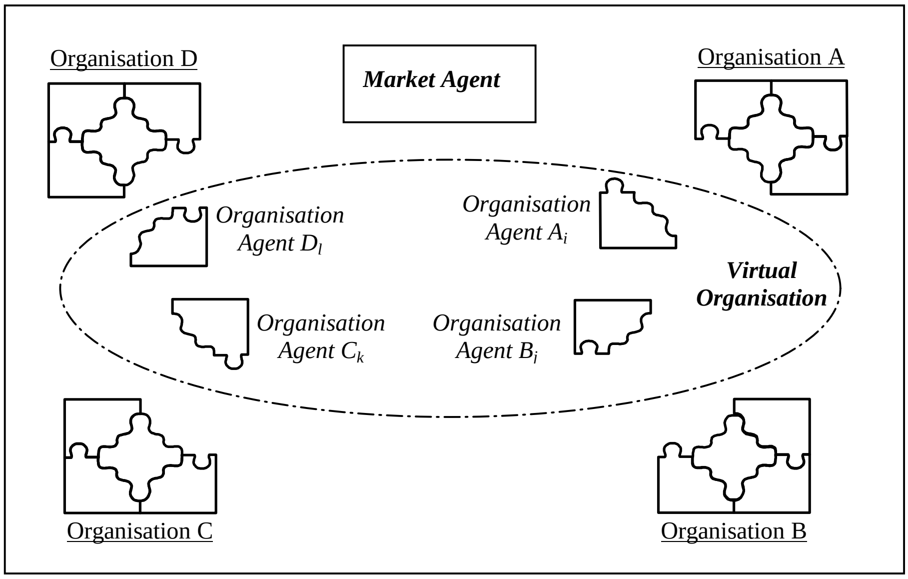

Figure 1 – Electronic Market architecture

2.1 Market Agent

When a specific consumer's need is identified (a consumer being either a human or an organisation), a new special agent, which is called **Market Agent**, is launched. The consumer selects a template which is configured according to the consumer' needs and evaluation preferences. This agent plays the role of coordinator in the Electronic Market where it is placed and its main goal is the Virtual Organisation formation. The Market Agent structure includes two principal modules: *"Goal Descriptor"* and *"Virtual Organisation Selector*" (see Figure 2).

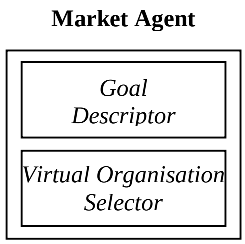

Figure 2 – Market Agent structure

We assume, for now, that the *"Goal Descriptor"* module is instantiated by the consumer. This module includes the description of the goal to achieve, which is decomposed in a set of sub-goals.
Each sub-goal (SGD) is described by means of a deadline and price value, and also a list of attributes. These attributes may not be defined by absolute values but instead they may be restricted to a set of preferable values in pre-defined domains. Values' preference is measured in a qualitative manner, based in the order by which attribute values are defined in the structure. Most favourable values (sub-domains values) should be defined as first elements of the ordered attributes' domain, and least favourable ones as last elements. Attributes' preference is defined in a qualitative manner similar to that of attribute values' preference, so attributes should be included in the structure description by its order of preference.
A sub-goal description is then represented by the following structure:

SGD = (SGId, D, LAt)

SGId: Sub-goal identification.
D: Deadline for sub-goal achievement.
LAt: List of evaluation attributes that (always) include the price as a default attribute. This list should be ordered by attributes' preference. Each element of the list has the following structure:

At_i = (AtId, Type, Dom)

AtId: Attribute identification.

Type: Identifies the type of the attribute values domain, that is, continuous or discrete.

Dom: Attribute values domain. These values are elements of a list ordered by preference. Each element of the list is either a single value (case of a discrete domain), or an interval of values (case of a continuous domain).

The sub-goals themselves may be related to each other through constraints attached to their attribute values. This is represented by a set of structures like the following:

constraint (SG, At, V, SG_c, At_c, V_c)

SG: Constraining sub-goal identification.

At: Constraining attribute identification.

V: Constraining attribute (domain) value(s).

The ‘c’ index in SG_c, At_c and V_c, refer to similar information concerning now the constrained issues.

The *“Virtual Organisation Selector”* module is responsible for the rational selection of the VO partners. In a first step, a set of invitations (one for every sub-goal described in *“Goal Descriptor”* module) is send to the Electronic Market system. The Market Agent second step leads to the evaluation of the received proposals. Senders of the most favourable ones are then selected and committed as partners of the VO. VO partners selection is done through multi-criteria evaluation and constraint-based negotiation protocols described later in this paper.

2.2 Organisation Agent

In order to be competitive in today’s economic markets, organisations need not only to be efficient in their business field, but also to be able to quickly react and adapt to new environments as well as to interact with other organisations. The control architecture adopted for the design of an organisation agent should meet these requirements. Figure 3 describes our proposed organisation agent structure.

Organisation Agent

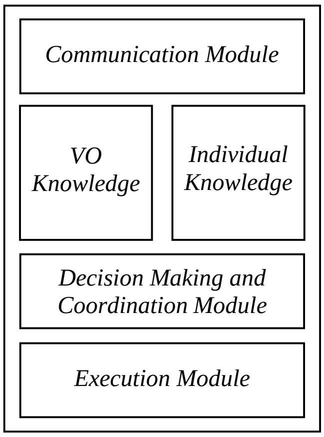

Figure 3 – Organisation Agent structure

An Organisation Agent’s structure comprises three functional modules: *Communication*, *Decision Making and Coordination*, and *Execution* modules, plus two knowledge-based modules: *Individual Knowledge* and *Virtual Organisation Knowledge* module.

The *“Communication module”* is responsible for all processes related with messages handling. Incoming messages are filtered and the ones that are not relevant for this specific agent are ignored. The relevant messages are then ordered by degree of importance. Outcome messages are sent out only to those agents that are known to be possibly interested in that particular piece of information. Agents use KQML - Knowledge Query Manipulation Language – (Finin, 1996) to exchange messages between themselves.

The *“Decision Making and Coordination module”* is the most complex one. Its role is to manage local tasks and cooperative behaviour. For our current presentation, the most relevant issue of this module includes the process to formulate a bid in response to an announcement (e.g., a proposal for becoming a VO partner). The bid formulation process uses a reinforcement learning technique to (re)formulate a response in each negotiation round. The learning process is done in a continuous way along the negotiation process. A detailed explanation of this process is given later in this paper (section 4.1).

The *"Execution module"* is the intelligent system responsible for the execution of local tasks an agent is committed to.

Information related to organisational and operational rules defined by the VO, is included in *"Virtual Organisation Knowledge"*. This module contains also information concerning individual organisation's rights and duties. This module is to be filled in during the VO formation phase, and can be modified during the VO operation as new organisations join the VO and partners leave the VO.

The *"Individual Knowledge"* contains information about the agent itself: agent own capabilities and current workload, as well as information related to other agents: other agents commitments and capabilities. Agent own capabilities are represented by the following structures:

has_capability (GId, D, LAt)

GId: Goal identification.
D: Relative deadline for goal achievement.
LAt: List of evaluation attributes. This information is similar to that included in the *Goal Descriptor* module of the *Market Agent* (see 2.1).

Each one of the agent's goals can constrain other goals (achieved by this one or other agents) by improving possible values to some attributes. This fact is mapped in a set of structures like the following:

constraint (SG, At, V, SG_c, At_c, V_c)

SG: Constraining sub-goal identification.
At: Constraining attribute identification.
V: Constraining attribute (domain) value(s).
The remaining parameters with 'c' index refer to similar information concerning now the constrained issues.

The Organisation Agent is a template supplied by the Electronic Market that, once being filled according to the individual organisation's specific tactics and strategies will represent that particular organisation in the market.

Although only the VO formation issue is being discussed here, the Organisation Agent structure proposed is able to support all phases of the VO life cycle.

3 NEGOTIATION PROTOCOL

Several examples of intelligent agent-based systems suitable for electronic market operations already exist (Chavez, 1996) (Tsvetovatyy, 1997). Moreover, electronic market functionalities available in existing systems vary from searching for goods to either simple purchasing or selling goods, or even to more complex negotiation of goods. It is this negotiation process between participants in an electronic market that becomes our main concern in this paper.

Automated negotiation systems have been using different approaches like auctions (Vulkan, 1998), game theory (Rosenschein, 1994) and intelligent agent technology (Jennings, 1996). Auction mechanisms are very popular due to its simplicity and well pre-defined rules, but only involve single criteria negotiations. Game theory mechanisms can only be applied in contexts with perfect information and perfect rationality. Intelligent agents technology is a more flexible paradigm suitable for dynamic and open environments, since agents can effectively cope with the complexity and large amount of information. Intelligent agents technology seems to be an appropriate paradigm to use in our case, since VO operates in very dynamic and complex environments.

A VO combines competencies derived from several heterogeneous and independent organisations present in the Electronic Market. A negotiation protocol has to be defined in order to select the individual organisations that, based on its capabilities and availability, will become members of the VO. In VO formation process, two important issues have to be considered:

- A multi-criteria evaluation is used to select the most favourable candidates: lower cost, higher quality, higher availability, etc, have to be taken into account.
- Organisation's competencies may be inter-related by constraints attached to their attributes' values, which leads to a kind of distributed constraints problem.

Attaching utility values to different criteria helps to solve the problem of multi-criteria evaluation. Generally, an evaluation formula is a linear combination of the current attributes' values weighted by their corresponding utility values (Oliveira, 1999) (Matos, 1998). However, in some cases, it can be a very difficult task to definitely attach

numeric values to attributes' utilities. A more natural and realistic situation is to simply impose a preference order over attributes' values and/or attributes itself. Contrary to the work mentioned above, we have adopted a multi-criteria evaluation based on a qualitative, not quantitative, measure.
Distributed Constraint Satisfaction Problems (Yokoo, 1992) are defined by a set of variables $(x_1,x_2,\ldots,x_n)$ which can take values from domains $D_1,D_2,\ldots,D_n$ respectively, and a set of constraints on their values. Here, in our scenario, variables are distributed among organisations in the Electronic Market and instantiations of the variables must satisfy these inter-organisation constraints.
In this paper we propose a multi-criteria negotiation protocol that uses Reinforcement Learning techniques for the selection of VO partners. We also include a simple algorithm for the resolution of constraint-based problems that can arise in this VO formation process.
Figure 4 illustrates the negotiation protocol. The Market Agent plays the role of coordinator in the VO formation process through its *"Virtual Organisation Selector"* module. This agent knows VO sub-goals that are described in the *"Goal Descriptor"* module. For each one of them, Market Agent sends out an announcement message to all organisations present in the Electronic Market. Organisations Agents reply with bid messages. Received bids are then evaluated using a multi-criteria function and the most favourable are selected. The negotiation process involves several rounds, and ends when a deadline is reached or when the *Market Agent* evaluates the current proposal(s) as acceptable. Those organisations that have sent out winning proposals become VO members and are committed to the respective goal achievement. Meanwhile, if different bids include attributes' values that are incompatible, the respective organisations' agents activate a negotiation process searching for an agreement through a set of compatible values.

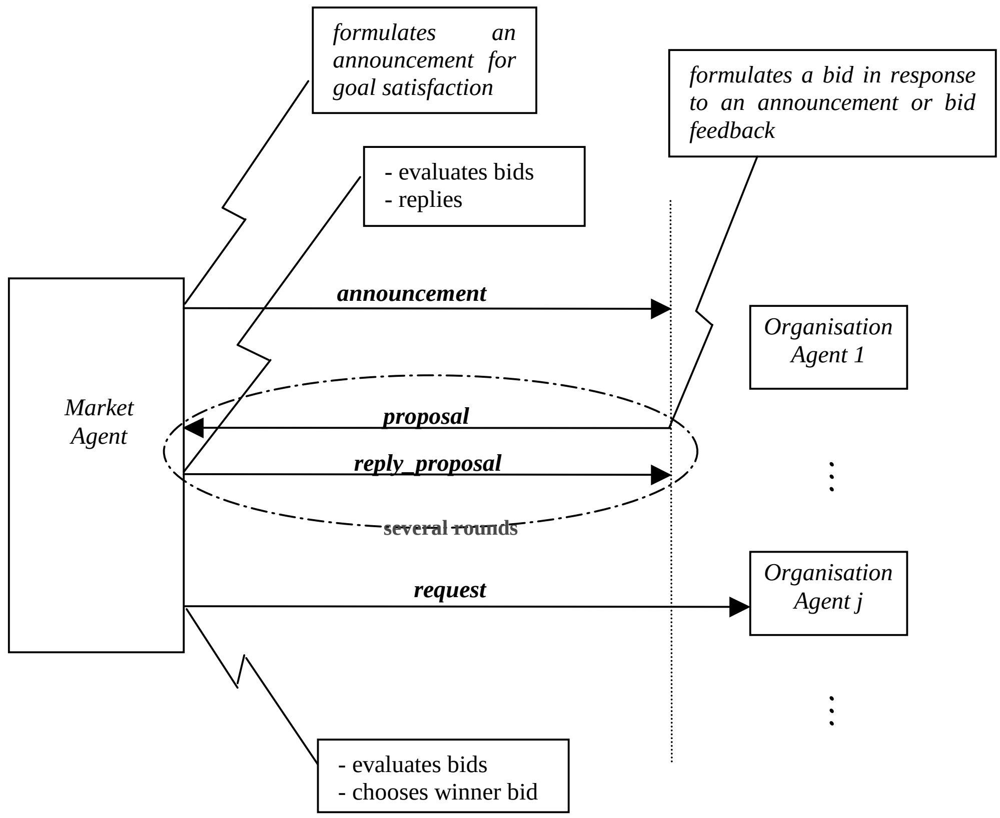

Figure 4 – Negotiation protocol

3.1 Announcement
The *Market Agent* sends out an announcement message for each of the VO sub-goals that are described in its *"Goal Descriptor"* module. The announcement, made by the *Market Agent,* will initiate the negotiation process. These messages are delivered to all organisations existing in the Electronic Market, and contain the following structure:

announcement (MAg, GId, D)

MAg: Market Agent identification.
GId: Goal identification.
D: Deadline value for goal achievement.

3.2 Bid formulation

In response to an announcement message or in response to a new negotiation round, the *Organisation Agent* formulates a bid based in its own capabilities, current availability, past experience (if possible) and *Market Agent* feedback. A reinforcement learning algorithm that enables continuous learning along the current negotiation process is used to formulate an agent bid. The message with the proposal is then sent out to the *Market Agent*, using the following structure:

proposal (OAg, PId, GId, T, LAt)

OAg: Organisation Agent identification.
PId: Proposal identification.
GId: Goal identification.
T: Time due for goal achievement.
LAt: List of attributes' values. Each element of the list has the following structure:
At_i = (AtId, V), where:
AtId: Attribute identification.
V: Attribute value.

Some constraints on other sub-goals' attribute values may result from accepting a particular agent's bid. This information is sent out to the *Market Agent* in one or possibly more messages including the following structure:

constraint (OAg, PId, G, At, V, G_c, At_c , V_c)

OAg: Organisation Agent identification.
PId: Proposal identification.
G: Constraining goal identification.
At: Constraining attribute identification.
V: Constraining attribute (domain) value(s).
The remaining parameters with 'c' index refers to similar information concerning now the constrained issues.

3.3 Bids evaluation

The *Market Agent* through its *"Virtual Organisation Selector"* module evaluates all bids received in the current negotiation round (for each one of the announced goals) based on a multi-criteria function. The *Organisation Agent* that presents the most valuable bid is informed that its bid is the selected one in the current negotiation round. The *Market Agent* then sends out a message to this organisation agent:

reply_proposal (PId, winning_round)

To other agents that participate in this round and are currently loosing, an advice in order to improve future proposals is also sent out:

reply_proposal (PId, loosing_round, Reason)

This message informs the respective *Organisation Agents* that *Market Agent* has received a better proposal, and their current bids are loosing the negotiation. Moreover, this message also includes the reason why this particular bid was not chosen, that is, what attributes were not considered good enough and how they can be improved.
After that, a new negotiation round begins and *Organisation Agents* will send new proposals to *Market Agent*, who will then reply accordingly. After a deadline is reached or a satisfactory proposal is received, the *Market Agent* will end the negotiation episode and organisations that have formulated the winning bids are informed that they are included in and committed to the new created VO:

request (GId, PId)

If an admissible bids' set cannot be found because some attributes values are mutually incompatible, a constraint satisfaction resolution process is activated. This process is described in the following section.

3.4 Constraint satisfaction problem resolution

Here, in our scenario, variables (describing attributes of goals) are distributed among organisations in the virtual marketplace and instantiations of those variables must satisfy existent inter-organisation constraints.

We propose a distributed constraint based algorithm that tries to obtain a global solution that satisfies all constraints and is optimal for the set of organisations included in it.

In response to the *Market Agent*, the *Organisation Agent* formulates a proposal in order to become a VO partner. This proposal message includes a list of attributes' values the organisation is willing to deliver according to its own self-interested objectives. If this particular proposal puts constraints on other sub-goals' attribute values, that same information is then sent to the *Market Agent*[1].

If the *Market Agent* receives proposals that include mutually incompatible attributes' values, it sends out that information to the respective constrained agents. The *Market Agent* job is to put in touch mutually constrained agents. It does not participate any more in the constraint problem resolution.

Constrained agents have to negotiate between them in order to agree in a set of admissible values for their constrained attributes. The initial proposal that a particular agent have made is its best local solution at that moment. We will call it solution $s*$. In order to solve the constraint problem and reach an agreement, agents change possible local solutions. Choosing the local solution s^k (necessarily different from the best solution $s*$) implies decreasing the agent' utility value, if compared to the best solution for the agent. We will call this difference the decreasing utility value (du^k), which can be calculated as the difference between the evaluation values[2] of either solutions:

$$du^k = Ev(s^*) - Ev(s^k)$$

Each local solution proposed by an agent to other(s) during the constraint resolution process should include the decreasing utility of that particular solution to that agent. Agents will agree in a final solution that presents the lowest global decreasing utility value (that is, the minimum sum of all constrained agents' decreasing utility). When this agreement is reached, the final solution for formulating an acceptable proposal is sent out to *Market Agent*.

Next we describe the procedure adopted in the resolution of the distributed constraint satisfaction problem:

(1) An agent receives a set of bids that are the most favourable for the senders. These sender agents also indicate the decreasing utility value of the proposed solutions. Among the solutions proposed, the receiver agent then selects the most favourable one according to its own objectives, and calculates its decreasing utility value. The receiver agent also calculates the decreasing utility values for the proposed solutions received, and sends out this information to sender agents.

(2) Receiver agent selects, within the solution space, the most favourable solution that has not been proposed yet. The local agent decreasing utility value for participating in this new solution is also calculated.

(3) Receiver agent chooses between solution obtained in (1), solution obtained in (2) and the better solution already proposed in the past. The solution that presents lower decreasing utility value is selected. Notice that we are comparing different decreasing utility values: total decreasing utility for solution (1) and local agent decreasing utility for solution (2). If the solution selected is the one already presented in the past or the one obtained in (1), then this is the final solution (the proposal that is mutually acceptable in accordance to the constraints). If the solution selected is the one obtained in (2), then it is sent out to constrained agent(s) participating in the negotiation and the constraint resolution process continues.

Usually, an organisation does not want to reveal its private information. In the procedure described above, not all possible variable domain values are made available since the beginning to other organisations in the system. Instead, this variable domain values is made available to others only when necessary, in order to reach an agreement, as the negotiation continues. Another aspect that we want to emphasise is the fact that solution utilities values, exchanged between organisations during the negotiation process, does not represent the real utility. Instead, agents exchange just utilities deviations (offset values).

Figure 5 illustrates how the distributed constraint based procedure is executed.

Suppose the existence of two organisations O_1 and O_2 and a VO objective described by means of goals G_x (with attributes A_{x1}, A_{x2}, …, A_{xn}) and G_y (with attributes A_{y1}, A_{y2}, …, A_{ym}). Organisation O_1 knows how to achieve goal G_x and Organisation O_2 knows how to achieve goal G_y. Possible values for attribute A_{x1} are represented as x_1, x_2, ..., x_5 (by O_1's order of preference). Possible values for attribute A_{y2} are represented as y_1, y_2, ..., y_6 (by O_2's order of preference).

[1] As a simple example, suppose that a VO has to be formed in order to deliver a specific type of car. The attribute *"dimension"* of the component *"wheel"* may restrict the values of the attribute *"material"* of the component *"screw"*.

[2] In section 4.2, we present a formula for calculating a bid (solution) evaluation value. For now, we will only refer that evaluation values are defined in the domain [0,1].

Attributes A_{x1} and A_{y2} are mutually constrained, as illustrated by the closed lines in figure 5. The evaluation value associated with particular solutions is enclosed in brackets.

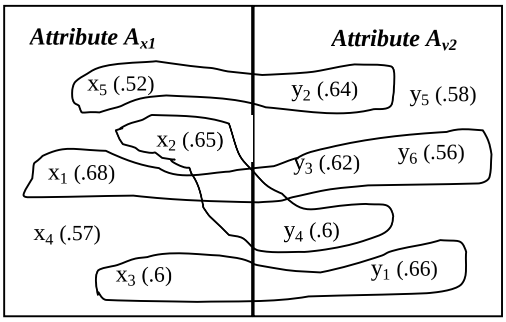

Figure 5 - Constraint space solution

Organisations O_1 and O_2 establish the following set of conversations:

- O_1 proposes
  - (A_1=x_1 , A_2=y_3) , O_1 du = 0
  - (A_1=x_1 , A_2=y_6) , O_1 du = 0
- O_2 evaluates solutions proposed by O_1, and sent out that information as a bid to O_1
  - (A_1=x_1 , A_2=y_3) , total du = 0.04 (0.04=0.66-0.62)
  - (A_1=x_1 , A_2=y_6) , total du = 0.1

  O_2 proposes
  - (A_2=y_1 , A_1=x_3) , O_2 du = 0
- O_1 evaluates solution proposed by O_2, and sent out that information as a bid to O_2
  - (A_2=y_1 , A_1=x_3) , total du = 0.08

  O_1 proposes
  - (A_1=x_2 , A_2=y_4) , O_1 du = 0.03
- O_2 evaluates solution proposed by O_1, and send that information as a bid to O_1
  - (A_1=x_2 , A_2=y_4) , total du = 0.09

  O_2 proposes
  - (A_2=y_2 , A_1=x_5) , O_2 du = 0.02
- O_1 evaluates solution proposed by O_2, and send that information as a bid to O_2
- (A_2=y_2 , A_1=x_5) , total du = 0.18

  O_1 accepts the solution already proposed in the past
  - (A_1=x_1 , A_2=y_3) , total du = 0.04

The final solution is the last proposal, which besides being in accordance with mutual constraints has the lowest total decreasing utility value.

4 DECISION MAKING FOR NEGOTIATION

4.1 Bid Formulation

Organisation Agents formulate bids for the goal satisfaction in response to an announcement message (or feedback messages from previous bids) sent by the *Market Agent*. The bid formulation process uses a Q-learning algorithm endowing the agent with the capability to learn on-line along the negotiation process. This is specially important when the agent is dealing with unexpected dynamic changes in the market. After each negotiation round, the *Market Agent* will reply to individual organisation agents by telling them how their bids could be improved in next rounds.

Reinforcement learning (RL) systems learn how to behave through trial-and-error interactions with its environment. Unlike supervised learning, no samples input-output pairs from the function to be learned are required, but instead the system receives a scalar evaluation from the environment for each action selected in a specific situation. RL is based in the idea of rewarding actions that produce good results and punishing those that produce bad results.

Q-learning is a reinforcement learning algorithm that maps state-action pairs to values, called Q-values. When an agent in current state *s*, performs action *a*, receives reward *r* and arrives at next state *s'*, the correspondent Q-value is updated as follows (Sutton, 1998):

$$Q(s,a) \quad = \quad Q(s,a) + \alpha\left(r + \gamma \max_{b} Q(s',b) - Q(s,a) \right)$$

where $0 \leq \alpha \leq 1$ (learning rate) and $0 \leq \gamma \leq 1$ (discount factor).

The discount factor is used to decrease the weight of reinforcements received in the future. It causes immediate reinforcements to have more importance than future reinforcements.

An agent is faced with a dilemma in order to choose what action to perform in each specific situation. Should it choose the most promising one? Or should it try to explore new states? If no exploration is made, the agent although using a greedy strategy, may miss the dynamics of the market. If too much exploration is made, the agent can make a lot of poor choices. It is necessary to ensure that sufficient exploration is made while choosing actions with high known Q-values (*Boltzmann exploration*):

$$p(a) \quad = \quad \frac{e^{Q(s,a)/t}}{\sum_{a' \in A} e^{Q(s,a')/t}}$$

where $t \geq 0$ (temperature)

p(a) is the probability of choosing action *a* and *t* is a parameter that controls the amount of exploration (it is usually decreased over time to decrease exploration). When a high temperature value is used, all actions are chosen with approximately equal probability. When a low temperature value is used, more highly evaluated actions are chosen.

We have adapted the Boltzmann exploration formula to our application domain, by not exploring all the possible actions but reducing the exploration to set A of those actions which are constrained by means of the *Market Agent* feedback to past bids.

In our application domain, a state is defined by the n-uple:

$s = (V_1, V_2, \ldots V_n)$, V_i is the value for attribute i

The state will also represent the bid sent to the *Market Agent*.

We define an action by the n-uple:

$a = (A_1, A_2, \ldots, A_n)$, , $A_i \in$ {increase, decrease, maintain}

Continuous attributes' values can be increased (decreased) by pre-defined amount steps. Discrete attributes' values can be increased (decreased) by moving to the next (previous) element in the enumerated domain.

When an *Organisation Agent* receives a feedback message from the *Market Agent* to its previous bid (this bid represents the state *s* that results from performing action a_p in state s_p), it tries to formulate a new bid following the procedure:

(1) Calculate a reward value for the previous bid (that also defines the current state *s*). This reward value is calculated using the qualitative evaluation included in the received feedback message:

$$\begin{aligned} r \quad &= \quad n && \text{, if winner} \\ &= \quad n - \sum_i penalty^i && \text{, if not winner} \qquad , 0 \leq penalty \leq 1 \end{aligned}$$

In the above formula, penalty is a parameter that decreases the reward value for bids with low evaluated values (remember that bids were evaluated by the *Market Agent*). This parameter should be defined by each organisation agent according to its own criteria and its value increases with the evaluations' category: low/high, very_low/very_lhigh, extremely_low/extremely_high. In this formula, attributes should appear in the same order they apper in the feedback message sent by the *Market Agent* (in this feedback message, attributes are ordered by *Market Agent*'s order of importance).

(2) Update the pair state/action Q-value ($Q(s_p,a_p)$) using the Q-learning formula presented above.

(3) Store the current sate *s*, and its associated reward value *r*.

(4) Derive all **new** promising actions *a'* taking into account that specific feedback message. For instance if the feedback message evaluates *attribute_2* as *too_low*, one possible promising action is to increase *attribute_2* and maintain all the others.
Store pairs *s/a'* with a default Q-value (*Q(s,a') = default*).

(5) Move to the state *s** with greatest reward value.

(6) Choose the action *a** with greatest probability value, according to the Boltzmann exploration formula, (note that actions not ever tried are stored in that state *s** with Q-value=*default*).

(7) Perform action *a** and send the new bid (state) out to the *Market Agent*.

4.2 Bid Evaluation

In order to select the most promising partners in the VO formation (or, in a more general electronic market environment, the most favourable buyers/sellers for a business transaction) a negotiation process is started by the Market Agent. This negotiation process comprises several rounds, starting when the *Market Agent* sends an announcement for all agents registered in the electronic market. The negotiation ends when a deadline is reached or a satisfactory proposal is received. At each negotiation round bids are evaluated, and the one with the greatest evaluation value is considered the winner in current round. Received bids comprise a solution to a specific needed goal, and include proposed values for multiple issues.
Bid evaluation is done through a multi-criteria function that encodes the attributes' and attributes values' preferences initially stipulated by the *Market Agent* and defined in its *"Goal Descriptor"* module. This multi-criteria evaluation function is defined by the following formula:

$$Ev = \frac{1}{Deviation}$$

$$Deviation = \frac{1}{n} * \sum_{i=1}^{n} dif(\mathrm{PrefV}_i, V_i)^i$$

where n = number of attributes for a specific goal.

In the formula above, each parcel should be presented in increasing order of preference. Attribute identified by number 1 (*i=1*) is the one that *Market Agent* classifies as the least important, and attribute identified by number n (*i=n*) is the most important.
The function $dif(PrefV_i,V_i)$ quantifies for an attribute *i*, the degree of acceptability of the current value (V_i) proposed by a specific organisation agent when compared to its preferable value *(PrefVi)* for the *Market Agent*. If the attribute values domain is of continuous values, this quantification simply is a normalised difference between the two values V_i and $PrefV_i$. If the attribute values domain is of discrete values, the result of the $dif(PrefV_i,V_i)$ function is now calculated as a normalised difference between the preference attached to V_i and $PrefV_I$, which can be calculated as the difference between the relative position of the two values in the enumerate domain values specification[3].
The winner bid in the current round is selected as the one that presents the highest evaluation value, since it is the solution that contains attributes values closer to the preferable values. The winner bid in this current round is compared with the winner bid in all past rounds, and the best one is selected.
All other bids are compared to the winner bid in order to inform the respective agents about the reason why their bids were not selected. A comparison is made for the values proposed for the most important attributes (from *Market Agent* point of view, of course). These values are then classified in a qualitative way: low/high, very_low/very_high, extremely_low /extremely_high.
Let us suppose that δ_i (i=1,2) splits the domain into the three mentioned regions, and *w* is the winner bid.

For all bids $k \neq w$, do
  For all most important attributes *i*, do
    $\Delta_{i,k} = \mathrm{dif}(V_{i,k}, V_{i,w})$
    If $\Delta_{i,k} < \delta_1$ then **Evaluation** = low (high)
    Else If $\Delta_{i,k} < \delta_2$ then **Evaluation** = very_low (very_high)
    Else **Evaluation** = extremely_low (extremely_high)

[3] In the *"Goal Descriptor"* module of the *Market Agent*, the domain values of a discrete attribute are defined in a list ordered by preference (see 2.1).

This information is included in the message that is sent out to all bidding agents except the currently winning agent, as a reason why their bids have not been selected. Such information is a list where each element is a pair (AtId, Evaluation), where AtId means the attribute identification. Agents will receive this information and will formulate new bids to send to *Market Agent* in the next negotiation round.
The negotiation process ends when:

- Market Agent receives a bid which has a satisfactory evaluation value (Ev). This bid is the winner bid.
- A deadline is reached. The winner bid is the one that presents the highest evaluation value among all bids received until then.

4.3 Preliminary work with Q-Learning

Our previous work with Q-learning was experimented in a electronic market where 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. 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, 1997).

One of the several experiments made (Oliveira, 1999b) includes adaptive agents, using different α values, and a "step changing price agent" (SCPA). We have studied the influence of parameter α on agent's adaptive behaviour. Figure 6 shows the performance of the agents using different alpha values (α=0.05, 0.1, 0.25, 0.5) when compared with SCPA. As expected, 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 α=0.25 obtains the best cash flow (utility) results. Initially, and after each SCPA change, the α=0.5 agent obtains the best results (highest slope on the cash flow 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.

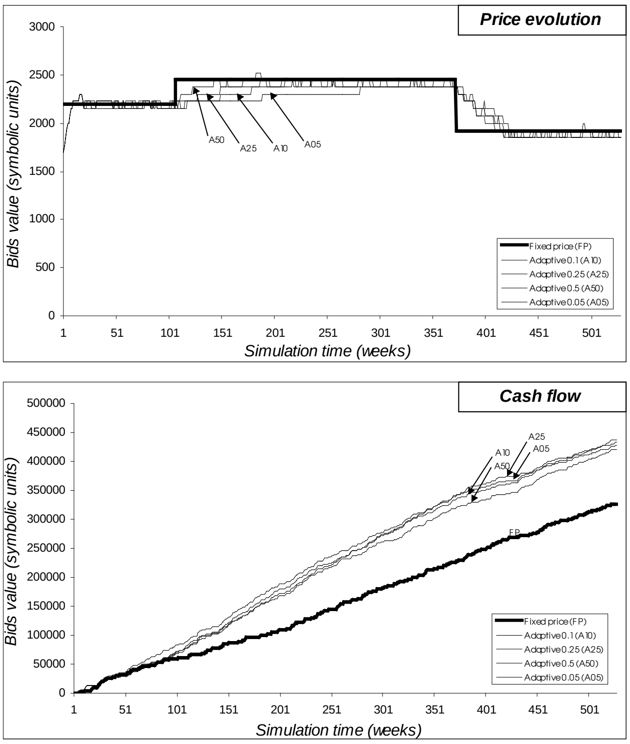

Figure 6 - Comparison between different adaptation coefficients.

We are confident that using the Q-learning techniques presented in previous sections, that includes not single criteria problems (like the results presented above) but multi-criteria problems will conduct to similar results.

5 CONCLUSIONS AND WORK IN PROGRESS

Virtual Organisations are of increasing importance mainly due to its flexibility, agility and efficiency. These main characteristics enable VO successful behaviour in dynamic and complex business environments.

In this paper we propose the use of intelligent agent technology for Electronic Commerce and VO formation.

In the KRAFT system (Preece, 1999) agent-based technology is also used to support VO formation and operation through a simple constraint fusion mechanism. KRAFT supports VO operation where members with heterogeneous knowledge and data models exchange information in the form of constraints. Instead, our proposal includes a more complex and powerful negotiation protocol suitable both for VO formation and Electronic Commerce in general. This new protocol includes resolution of multi-criteria and distributed constraint-based problems. We also show how agents can efficiently compete in a dynamic market by adapting themselves with an appropriate algorithm (Q-Learning) to market changing, due to their continuous on-line learning capabilities.

A Multi-Agent System architecture was also proposed as a support for the implementation of the Electronic Market where individual business partners meet each other, negotiate and cooperate.

A prototype featuring all the mentioned characteristics is being developed for future experimentation.

ACKNOWLEDGEMENT

We would like to acknowledge implementation efforts of Henrique Cardoso and exchange of ideas with José Manuel Fonseca.

REFERENCES

1. Bakos Yannis. The Emerging Role of Electronic Marketplaces on the Internet. Communications of the ACM, Vol. 41, No. 8, pp. 35-42, August 1998.
2. Chavez Anthony, Maes Pattie. Kasbah: An Agent Marketplace for Buying and Selling Goods. Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi Agent Technology, London, UK, April 1996.
3. Faisst Wolfgang. Information Technology as an Enabler of Virtual Enterprises: A Life-Cycle-Oriented Description. Proceedings of the European Conference on Virtual Enterprises and Networked Solutions. Paderborn, Germany, April 1997.
4. Finin Tim, Labrou Yannis, James Mayfield. KQML as an agent communication language. Software Agents. Jeff Bradshaw (Ed.), MIT Press, 1997.
5. Fischer Klaus, Muller Jorg, Heimig Ingo, Scheer A-W. Intelligent Agents in Virtual Enterprises. Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi Agent Technology. London, UK, April 1996.
6. Guttman Robert H., Moukas Alexandros G., Maes Pattie. Agent-Mediated Electronic Commerce: A Survey. Knowledge Engineering Review, June 1998.
7. Jennings N. R., Farantin P., Johnson M. J., O'Brien P., Wiegand M.E. Using Intelligent Agents to manage Business Processes. Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi Agent Technology. London, UK, April 1996.
8. Matos Noyda, Sierra Carles. Evolutionary Computing and Negotiating Agents. Proceedings of the Workshop on Agent Mediated Electronic Trading, Second International Conference on Autonomous Agents. Minneapolis, USA, May 1998.
9. O'Hare G. M. P., Jennings N. R., Foundations of Distributed Artificial Intelligence, John Wiley & Sons, 1996
10. Oliveira E., Fonseca J. M., Garçã, A. S. MACIV - A DAI Based Resource Management System. International Journal on Applied Artificial Intelligence, Vol. 11, pp. 525-550, 1997.
11. Oliveira Eugénio, Fonseca José M, Steiger-Garção A. Multi-criteria negotiation on Multi-Agent Systems. Proceedings of the First International Workshop of Central and Eastern Europe on Multi-Agent Systems, St. Petersburg, Russia, June 1999.
12. Oliveira Eugénio, Fonseca José M, Jennings Nick. Learning to be competitive in the Market. American Association of Artificial Intelligence Workshop on Negotiation, Orlando, USA, July 1999b.
13. Preece Alun, Hui Kit, Gray Peter. KRAFT: Supporting Virtual Organisations through Knowledge Fusion. (in this same volume)
14. Rosenschein Jeffrey S., Zlotkin Gilad. Rules of Encounter. MIT Press, 1994.
15. Sutton R. S., Barto A. G. Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA, 1998.
16. Tsvetovatyy Maksim, Gini Maria, Mobasher Bamshad, Wieckowski Zbigniew. MAGMA: An Agent-Base Virtual Market for Electronic Commerce. Journal of Applied Artificial Intelligence, Special issue on Intelligent Agents, Vol. 11, No. 6, 1997.
17. Vulkan Nir, Jennings Nick R. Efficient Mechanisms for the Supply of Services in Multi-Agent Environments. Proceedings of the First International Conference on Information and Computation Economics. Charleston, South Carolina, October 1998.
18. Yokoo Makoto, Durfee Edmund, Ishida Toru, Kuwabara Kazuhiro. Distributed Constraint Satisfaction for Formalizing Distributed Problem Solving. Proceedings of the Twelfth IEEE International Conference on Distributed Computing systems, June 1992.