A Computation Environment for Automated Negotiation: A Case Study in Electronic Tourism

Alan Silva
Federal University of Alagoas
Computation Institute
Alagoas - Brazil
alan@tci.ufal.br

José Alencar Neto
Federal University of Alagoas
Computation Institute
Alagoas - Brazil
alencar@tci.ufal.br

Ig Ibert
Federal University of Alagoas
Computation Institute
Alagoas - Brazil
ibert@tci.ufal.br

ABSTRACT

The automated negotiation topic plays an important role in e-commerce research. However, despite considerable work on automated negotiation, few research efforts have aimed at software engineering facilities such a reuse and flexibility. To address this issue, we propose a novel computation environment for building agents with flexible negotiation strategies to function in various virtual business domains. Regarding the negotiation strategies, some decision taking assistance techniques may be used in group, e.g. rule-based reasoning and techniques to machine learning. Considering the flexibility of acting in various business domains, a specific agent can be programmed to work in as many business domains as necessary, and essentially, it will also be possible to re-configure the agent´s business domains at execution time. All the experiments in this work were idealized in the terms of a tourism negotiation package. Results from these experiments demonstrate effectiveness of our proposal.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Intelligent agents, Multiagent systems*; I.2.6 [**Artificial Intelligence**]: Learning; J.4 [**Computer Applications**]: Social and Behavioral Sciences—*Economics*

General Terms

Economics, Experimentation.

Keywords

Electronic commerce; Trading agents; Agent and multiagent architectures; Cognitive models.

1. INTRODUCTION

Nowadays, one of the main agent-based software engineering application domains is the use of agents for automating negotiation processes. The aim which is objectified by the major researches in the automated negotiation subject is to make its processes as similar as possible to the traditional commerce.

Generally, the problems of automated negotiation are complex and their solutions are sophisticated, demanding some subjects to be meditated and solved in an integrated way, such as: (1) What are the possible actions a certain agent can take in behalf of a company or an individual?(2) When these actions may or shall be executed?(3) Which vocabulary are these agents going to discuss on?(4) How will the agents infer to decide on a certain action considering the best option for a particular situation on the negotiation?(5) What is the virtual domain the machine is inserted on?

The currently available solutions have specific negotiation strategies which are related to a determined domain. When it comes to practical usage, negotiation strategies have been developed for very specific negotiation environments. This leads to solutions with little flexibility to other negotiation environments adaptation.

Moreover, negotiation environments have been very restricted to a specific negotiation model, normally based on auctions. These environments usually have many restrictions, making an agent's implementation very connected to a definitive negotiation model.

Another current problem is the fact that the negotiation strategies are connected and encapsulated to programming languages. The way an agent interacts in a negotiation process normally is implemented directly inside of it, along with questions related to agent's physical location and communication infrastructure and its adaptation to a determined negotiation environment.

At last, the existing negotiation mechanisms have very static characteristics. For instance, assuming that the agent was already programmed under some mechanism to approach its opponents, this mechanism will be exactly the same in all negotiation moments. It is not able to be reused in other mechanisms.

To address the issues above, in this paper we present the design and implementation of an environment to support automated negotiations, offering support for the use of various existing negotiation models, together with their respective negotiation strategies. This way, it is possible to integrate different models, approximating the automated negotiations to the way the real world works.

Thereby, we propose a flexible and open architecture for the negotiation environment, where the number of buyers and sellers, or the offer and demand for services or products, can be changed during execution time. In a flexible

SAC'07 March 11-15, 2007, Seoul, Korea

way, the negotiator agent can grow in the number of negotiation strategies and also in the number of business domains it is involved. These characteristics were implemented in the proposed environment using ontology concepts and production rules.

The use of ontology enabled the negotiator agents to be implemented, a priori, for any type of business domain. The fact is that the environment supplies a protocol for the agents to interact on the same ontology, allowing an agent to be detached from the various business domains it may possibly come to negotiate in. The individuality of each negotiator agent is therefore tied to the rules that he possesses.

The remaining sections of this paper are organized as follows: Section 3 describes the proposed environment. Sections 4 and 5 present, respectively, the applied case study and experiments. Section 2 describes related works. Finally, a conclusion and future works of this work is discussed in Section 6.

2. RELATED WORK

The traditional solutions [3, 2, 4, 5] try to solve the problems related automated negotiation individually and therefore they do not contemplate the process of negotiation as a whole. In other words, the existing solutions tend to observe a restricted aspect of the complete negotiation process, keeping relevant solutions from being used in an unified way, because they are solutions created "by request" for a certain context [7].

In [7, 1, 4] the authors presents significant progresses for automated negotiation in e-commerce in terms of problems above mentioned. However, some other aspects should be considered for a higher proximity to negotiations in the real world.

First, in Bartolini [1], an open automated negotiation environment is proposed. This environment is based on central negotiation mechanisms, which controls all the negotiation environment operation, including the necessary conditions to accomplish an agreement. The disadvantage of this approach is that the conditions in which the negotiator agent, saller or buyer, has to accomplish an agreement can be very individualist and extremely variable concerning to time.

Besides that, in the work proposed by Bartolini [1], there are no existing support for the construction of negotiator agents, the truth is that the agent´s planners will have to know the rules of negotiation and implement them directly inside their agents, for every possible negotiation mechanism.

In addition, as the conditions for accomplishing an agreement are all predefined and there only exists a single, and previously specified, negotiation object that is involved in all negotiations, the bargain power among the involved negotiating agents is diminished. Thus, it will not be possible to negotiate a product specification in different negotiation conditions.

Tamma [7] tries to reduce this coupling problem between a certain negotiator agent and a specific negotiation domain using ontologies. This work proposes to attribute the responsibility of modeling all the negotiation protocols to a single ontology. This way, as the marketplace is entered, its agents shall be able to negotiate without any necessary type of previous adjustments. The problem is that, in case there is a marketplace with a negotiation protocol different from the foreseen in the ontology, the agent will not be capable of negotiating because a previous implementation was not made. The second problem is that for business domains that need the power of a larger expressiveness (with many terms and different intentions) this single ontology would not be cohesive, as it would have to deal with many non-related subjects.

In Oliveira [6] there was a great progress in determining an environment and in the automatic learning for negotiators, but no subjects related to how an ontology mechanism can provide easy expressiveness and interaction between agents, of any type, were proposed.

3. ENVIRONMENT DESCRIPTION

At first an automated negotiation is seen as a game where the players, the agents, interact with each other aiming to reach one agreement. Figure 1 represents a negotiation abstraction. In the same figure we can observe two negotiator agents, which are side by side, trying to reach an agreement through the interactions. These interactions may occur in several forms, as well as the interactions between human negotiators.

Besides, it is very often that during the negotiation process it maybe necessary to gather information about the external environment. The interaction among the agents can bring more details and knowledge level, increasing, therefore, a negotiator's chance to reach a better success on their objectives. For instance, this knowledge may represent the quotation of petroleum barrels, the quotation of foreign currencies or even the weather forecast.

For this, the information on a business domain must be shared between the negotiators so that they can understand each other in a syntactic and semantic way. Hence, it was necessary to use ontologies for sharing this type of knowledge.

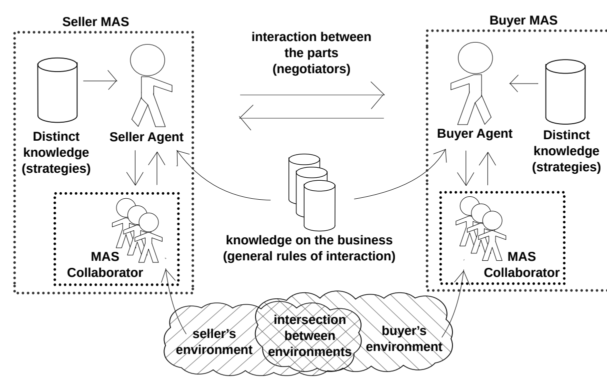

Figure 1: Negotiation Environment Abstraction

3.1 Negotiation Protocol

For communication purposes between negotiator agents, and in addition to a physical communication environment, it is necessary for these agents to knowledge under what circumstances and/or terms the negotiators will deal. The environment proposed in this work considers that all participant negotiating agents are well behaved, which means that the agents will not act out of the script. In other words, they will not send any message apart from the patterns already established inside the ontology. Therefore, we propose a protocol for negotiations, which simplifies the way new negotiation models and other virtual domains are inserted in the environment. This protocol is expressed in one ontology as illustrated by Figure 2. The main entities involved in the protocol are described as follows:

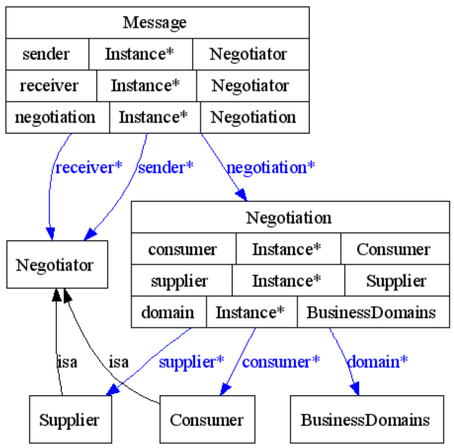

Figure 2: Ontology for describing the negotiation protocol

- Messages: The messages represent the pattern to information exchange between negotiator agents inside the negotiation environment. As described in Figure 2, a message can contain: the sender of the message, the address destination, a text content (*String*) and the negotiation model which will be used by the agents. The whole message will somewhat represent a specific term describing the business domain in action.

- Negotiation: In spite of providing the proposed environment the opportunity for the negotiator agent to act simultaneously in several negotiations, with several negotiators and in various business domains, the negotiations carried out, between two or more agents, are dealt in an individualized and bilateral way. In other words, a negotiation represents a messages exchange among just two negotiating agents, being generally abstracted for the other existing concurrent negotiating agents. This behavioral characteristic is given due to the necessity of creating a simplified model that is able to support a large number of negotiation modalities. Hence, in case of sharing relative information about current parallel negotiations from the same domain, it is necessary for the domain ontology to support such characteristic. This way, a negotiation is represented by a supplier and a consumer linked by a business model.

3.2 Negotiator Agent Architecture

The architecture of a negotiator agent is represented by Figure 3, where the components of each entity is highlighted, which are related to decision making and learning. The objective here is to show the framework structure for creating SMA Negotiator, as intelligent agents, that might use the available Artificial Intelligence techniques.

Understanding the term "knowledge" as the group of information that will influence the negotiator in its actions, a negotiating agent has different and well-defined knowledge, as seen on Figure 3.

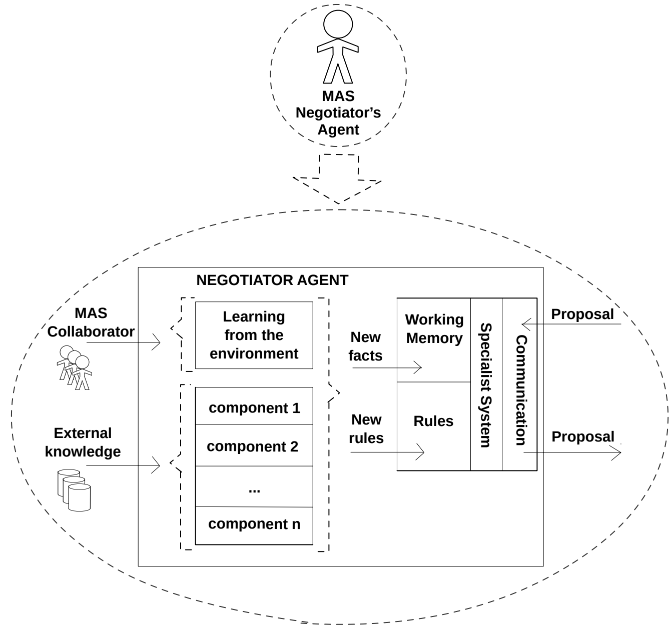

Figure 3: Negotiator Agent Architecture

- Environment knowledge: here is represented all the knowledge about the negotiator agent environment and everything that is useful to it;

- Component: It is the encapsulation of machine learning techniques (Case-based Reasoning, Neural Networks, Q-Learning, Data Mining, etc), used by the negotiator for learning purposes.

- Work Memory: This is where the facts gained from the environment, components and rules are found. In general, new facts can unchain new actions to be taken.

- Rules: Here are the basic rules for negotiating (configured *a priori*) and also the new component-added rules, which were inferred through the use of Artificial Intelligence.

4. CASE STUDY

This section describes the case study which was elaborated to test the environment proposed in Section 3. We elaborated an ontology of meanings and terms related to the actions and intentions for negotiating travel packages.

First, it is important to highlight that the ontology represents a negotiation that involves two types of agents: the customer agent and the virtual agency agent. In this negotiation two types of information are exchanged between the agents. First of all, the client agent requests a travel package, then the agency agent answers back with the requested travel packages.

Source 1: Type of the TravelPackage Message

```
(deftemplate TravelPackage
    (slot id)
```

```
        (slot agency)
        (slot trip_package_request)
        (slot transport_service)
        (slot lodging_service)
        (slot status)
    )
```

Source 2: Type of the TravelPackageRequest Message

```
(deftemplate TravelPackageRequest
        (slot id)
        (slot solicitor)
        (slot transport_option)
        (slot lodging_option)
)
```

Initially, the part of the ontology that regards to a travel package request, as represented in Figure 4, has the following terms:

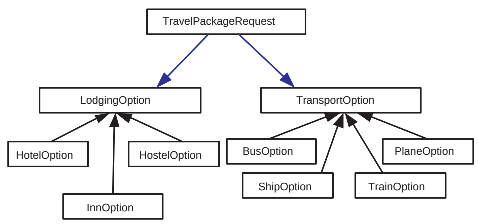

Figure 4: An ontology concerning a trip package request.

- *TravelPackageRequest*: This is the model for a travel package request that both agents must be aware of. Thereby, a client shall send his lodging preferences, the transportation type and other general preferences. Furthermore, these preferences are represented by other classes of the ontology: *LodgingOption*, *TransportOption* and *HotelOption*;

- *TransportOption*: Regarding the transportation options, information such as the city of origin, the city of destination, the amount of tickets and the departure shift were taken in account. Besides, the class also has subclasses that specifies the type of transportation: *BusOption*, *ShipOption*, *PlaneOption* and *TrainOption*;

- *LodgingOption*: Beneath, the lodging preferences are defined. There are three lodging types: *HotelOption*, *HostelOption* and *InnOption*.

In the same way, if an agency agent wants to send a travel package, it should know the part of the ontology that regards to the travel package (illustrated in Figure 5), that possesses the following terms:

- *TravelPackage*: A travel package represents a group of services that a certain travel agency will provide to a customer. For this case study purposes, only two services are provided: *LodgingService* and *TransportService*, as detailed beneath:

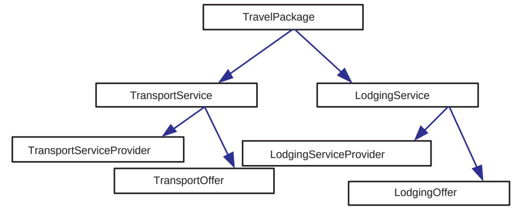

Figure 5: An ontology concerning a travel package.

- *LodgingService*: According to a specific lodging service request, the agency agent will describe the lodging service provider *LodgingServiceProvider* to the client, together with a possible *LodgingOffer* offer.

  - *LodgingService*: This way, an offer corresponds to the possibility of a business with a certain service provider. So when an agency agent receives a proposal it will check with its lodging service providers *LodgingServiceProvider* for somebody who may assists the given request. The lodging service providers will answer the agency agent with several related offers, allowing the agency agent to choose between the most convenient offer and define it in the respective lodging service *LodgingService*. Actually, this search for proposals is advisable to be delegated to a support agent, due to the fact that it may always be trying to discover new interesting offers. It is also important to notice that agency agents which possess the most attracting offers shall offer them to their clients;

  - *LodgingServiceProvider*: The lodging service providers *LodgingServiceProvider* are represented by hotels, hostels or housings able to provide their service and assist the offered demand made by the client agents to the virtual agencies.

- *TransportService*: As well as the lodging service, the transport service corresponds to the form that a certain agency will offer the transportation service to its customers. Therefore, the *TransportOffer* has a transportation offer *TransportOffer* and a respective transportation service provider *TransportServiceProvider*.

  - TransportOffer: The transportation offer refers to a certain service request. Thereby, for a travel request, there may exist various service providers able to assist such a request. Therefore, the agency agent shall choose the most convenient option to assist its customer´s request;

  - *TransportServiceProvider*: The transportation service providers *TransportServiceProvider* are the airlines, road and sea travel companies, that can offer the transportation to the customers and attend to the requests from the clients to the virtual agencies.

5. EXPERIMENTS

The experiments were made with a client agent that negotiated two travel packages with four distinct agency agents.

	A	B	C	D
Lodging	8907,75	8953,87	8917,75	8998,87
Transport	988,50	994,25	988,50	999,25
Lodging	8815,50	8907,75	8835,50	8997,75
Transport	977,00	988,50	977,00	998,50
Lodging	8723,25	8861,62	8753,25	8996,62
Transport	965,50	982,75	965,50	997,75
Grand Total	**9688,75**	9844,37	9718,75	9994,37

Table 1: Values bargained by the agency agents for each negotiation round inside the scenario 1.

The agency agents used negotiation strategies mapped as production rules, and they set the price according to the final value, adding the number of passengers and the luggage weight prices.

In the first round of negotiation, the client agent request was to reserve 15 tickets for passengers, with a total weight of 4.500kg for the luggage. Based on this information, the agencies could negotiate on several rule-based strategies, since the grand total of the package would surpass $10.000,00 and there would also be a luggage excess (this was an estimated value in order to trigger some rules).

The second request was a simple travel package, were there would be a negotiation for the final travel price and no luggage weight excess. Still, we supposed that the customer had a big travel history, and therefore the agency agent that used loyalty-based strategies would be able to have a more powerful bargain, which in this case happened only to one out of the four agency agents.

The results of both negotiations may be found on Table 1 and 2, representing the first and second requests.

	A	B	C	D
Lodging	99,85	99,92	99,85	99,92
Transport	99,85	99,92	99,85	99,92
Lodging	99,70	99,85	99,70	99,85
Transport	99,70	99,85	99,70	99,85
Lodging	99,55	99,77	99,55	99,77
Transport	99,55	99,77	99,55	99,77
Grand Total	199,00	199,54	**199,00**	199,54

Table 2: Values bargained by the agency agents for each negotiation round inside the scenario 1.

As we can see, in the fist negotiation round (table 1), the agent A easily won the negotiation due to its higher bargain-power possession based on the final package price, the luggage weight and, mainly, on loyalty. In the second negotiation (table 2), the agent C won for sending his proposal first.

The agent C, in the second negotiation, sent the same proposals as agent A, but, for having a smaller processing task, it sent its proposal faster than agent A, thereby winning the negotiation.

6. CONCLUSIONS AND FUTURE WORKS

In this paper we presented an environment for automated e-commerce using ontologies and production rules. The focus of the proposed environment is to give support for the development of computational negotiating agents that may act in a flexible way in negotiation environments, taking in to account that a negotiator agent may be projected to act in several negotiation environments.

A case study was presented in the domain of virtual travel agency. It showed that the use of rules as a mechanism for representing knowledge for an agent´s objectives, together with their way of representing the negotiation strategies, turns the agents flexible to changes in their strategies or in their objectives, compared with the agents described in Tamma [7]. Hence, the proposed work described in this paper shows that the business domains and its vocabulary restrictions are independent from the negotiation protocol.

In the evaluation of the Bartolini [1] model, improvements have been obtained taking in consideration the autonomy of the agents in the business environment. The proposed model gives each agent the ability to individually decide whether *(i)* to send or *(ii)* to accept a proposal.

As future work, we plan to apply the proposed environment to new business domains. This will be fundamental to improve the flexibility provided by the environment through the observation of specific applications requirements that may be widespread and eventually become part of the proposed environment structure.

7. ADDITIONAL AUTHORS

Hyggo Oliveira: Federal University of Campina Grande, Department of Electrical Engineering, Campina Grande, PB, Brazil. E-mail: (hyggo@dsc.ufcg.edu.br). **Angelo Perkusich**: Federal University of Campina Grande, Department of Electrical Engineering, Campina Grande, PB, Brazil. E-mail: (perkusic@dee.ufcg.edu.br). **Evandro Costa**: Federal University of Alagoas, Computation Institute, Maceio, Alagoas, Brazil. E-mail: (evandro@tci.ufal.br). **Luis Henrique Caetano**: Federal University of Alagoas, Computation Institute, Maceio, Alagoas, Brazil. E-mail: (lhtc@tci.ufal.br). **Elvys Soares**: Federal University of Alagoas, Computation Institute, Maceio, Alagoas, Brazil. E-mail: (elvys@tci.ufal.br).**Romulo Oliveira**: Federal University of Alagoas, Computation Institute, Maceio, AL, Brazil. E-mail: (romulo@dsc.ufcg.edu.br).

8. REFERENCES

[1] C. Bartolini, C. Preist, and N. R. Jennings. A Software Framework for Automated Negotiation. In *Software Engineering for Large-Scale Multi-Agent Systems*, volume 3390 of *Lecture Notes in Computer Science*, pages 213–235. Springer-Verlag, 2004.

[2] A. Chavez and P. Maes. Kasbah: An Agent Marketplace for Buying and Selling Goods. In *First International Conference on the Pratical Application of Intelligent Agents and Multi-Agent Technology*, pages 75–90, London, UK, 1996.

[3] P. Faratin, C. Sierra, and N. R. Jennings. Negotiation Decision Functions for Autonomous Agents. *Robotics and Autonomous Systems*, 24(3-4):159–182, 1998.

[4] P. Faratin, C. Sierra, and N. R. Jennings. Using Similarity Criteria to Make Issue Trade-offs in Automated Negotiations. *Artificial Intelligence*, 142(2):205–237, 2002.

[5] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, M. Wooldridge, and C. Sierra. Automated Negotiation: Prospects Methods and Challenges. *Group Decision and Negotiation*, 10(2):199–215, 2001.

[6] R. N. Oliveira, A. P. Silva, I. Bitencourt, H. Gomes, and E. B. Costa. A multiagent based framework for supporting learning in adaptive automated negotiation. In *accepted for publication at the 8th International Conference on Enterprise Information Systems - ICEIS 2006*, Paphos - Cyprus, May 2006.

[7] I. D. V. Tamma, S. Phelps and M. Wooldridge. Ontologies for supporting negotiation in e-commerce. *Engineering Applications of Artificial Intelligence*, 18(2):223–236, 2005.