REPUTATION AND TRUST IN THE CONTEXT OF LOGIC-BASED ARGUMENTATION

Paulo Novais, Cesar Analide, José Machado and José Neves
Universidade do Minho
Departamento de Informática
Campus de Gualtar, 4710-057 Braga, Portugal
Voice: +351 253 604466/70 Fax: +351 253 604471
{pjon, analide, jmac, jneves}@di.uminho.pt

Abstract – The specification of argument-based negotiation in e-Commerce environments aims not only at the definition of the best dealing strategies to be used by any set of computational entities in such environments, but also to mimic in its minds the reasoning processes of their human peers. Reputation may be one important parameter to be taken into account in the process of trust evaluation in a negotiation, once it can be analyzed either in terms of the quality of knowledge it carries into the negotiation process, or the referentials it may impose. In this work, such a model will be materialized by a sound syntactic and semantic tool for logic-based argumentation, where entities called agents and multiagent systems set the computational methodology to be pursued in the process of problem solving.

Keywords: Agents, Reputation, Trust, Negotiation, Argumentation.

I. INTRODUCTION

Experimental research in the fields of Computer Science (CS), Artificial Intelligence (AI) and Multiagent Systems (MAS) foresee an approximation of these disciplines and those of Social Sciences, namely Anthropology, Sociology and Psychology; much work has been done in terms of the humanization of the behaviour of virtual entities, by expressing human like feelings and emotions [1] [2] [3]. For example, work presented in [4] [5] detail studies and propose methodologies to assign emotions to machines. Indeed, an important motivation to the development of this work has its roots either in the efforts that are being made to enforce new forms of knowledge representation and reasoning in terms of an extension to the language of logic programming (i.e., the Extended Logic Programming (ELP) language [6] [7]), or how to follow up the road to assign emotions to machines. On the other hand, the use of incomplete information and the enforcement of exceptions to the extensions of predicates that make the bulk of the agents or multiagent system body of knowledge, in order to characterize their behaviour, is in itself another justification for the adoption of these formalisms in this knowledge arena.

The use of logic enables systems to be modelled with the added benefit of mathematical notation and non-ambiguity. Logic programming tools even provide a working prototype for the modelled system, amazingly reducing the time between formal specification and prototype development and testing.

Therefore, a formal model to humanize the behaviour of such entities has to be built, that will enforce new forms of knowledge representation and reasoning, on the road to assign emotions to machines.

II. MULTIAGENT SYSTEMS

In the current economical context, characterized by the existence of a global society, where information retrieving is crucial for any economical and social development, there are important technological challenges. The representation, maintenance and querying of information is a central part of this context and some questions may be answered: How can we obtain the adequate information at the adequate time? How can we supply the correct items for the correct people at the correct time? How and where can we get the relevant information for a good decision making? The organizations focus their competences in strategical areas and use external supplies, cooperating with sporadical partners, with the objective of reducing costs, risks and technological faults and, on the other hand maximizing benefits and business opportunities. One of the most radical and spectacular changes is the information unmaterialization, the task or procedure automation, the recourse to decision support systems or intelligent systems and to new forms of celebrating contracts (it is possible to practice commercial acts and celebrate deals using autonomous and pro-active computational agents). The e-Commerce has now new challenges, searching for new answers to old questions. The negotiation using electronic means and the electronic commerce framework direct to new forms of engagements with valid contracts, establishing contacts and negotiations between virtual entities [8]. This project aims to describe, analyse and validate open and global heterogeneous entities, called agents. They are computational entities, with a rich knowledge component, having sophisticated properties such as planning ability, reactivity, learning, cooperation, communication and argumentation. The use of the agent figure is particularly adequate to such problems. The objective
is to build logical and computational models, as well as implementing archetype with those entities, interacting in an open and global environment, with respect to the law norms (i.e., legislation, doctrine and jurisprudence) [9]. Agent societies may mirror a great variety of human societies, such as commercial societies with emphasis to behavioural patterns, or more bureaucratic ones, with pre-defined roles, obligation, contractual and specific communication rules. The entity models and behavioural patterns are verifiable.

The traditional programming languages do not support the description of certain types of action which usually involve computational agents. In genesis, systems that incorporate those functionalities have a multi-layer approach where, in first place, the system has to be specified from esoteric software sub-systems, network protocols, window systems, and so on... In second place, once one deals with multi-agent systems, it must be considered that those may answer to different and simultaneous demands, in a secure and error free way. An agent understands (observes and reasons), interacts with the environment, has its beliefs, desires and intentions.

These systems are reactive and are characterized by the possibility of moving actions that interact with external worlds. In terms of knowledge representation, the organizational or ecological model has been selected, not only to combine the potentialities of epistemic systems, but also to add a social component. In terms of formalization, the classical logic, that follows the temporal and platonic notions of truth (i.e., classical logic embraces a global, monotonic and static truth concept where a formula can only have one of two values, true or false, that is independent of the time point where the evaluation is computed), is impracticable for the representation of dynamic systems with state variation.

An agent may be able to manage its knowledge, beliefs, desires, intentions, goals and values. It may be able also to plan, choose, receive information or instructions, or react to environment updates. It can communicate with other agents, share knowledge and beliefs, and respond to the others agent requests. It can cooperate, diagnosing errors or information faults in its knowledge bases, sharing resources, avoiding undesirable interferences or joining efforts to revisit the knowledge bases in order to reach a common goal. Each agent is a system of subagents made up by possibly concurrent entities that are function oriented and specialized and that execute several goals and instructions following the agent orientation (e.g., each one of the functionalities: reactivity, reasoning, belief revision, argumentation, justification, learning, dialogue, information retrieving, preference evaluation, strategy and diagnostic). The subagents are coordinated by a meta-layer that guarantees the internal task distribution, the knowledge revision, the inter-subagent communications, the decision making and the possible interaction with the external world (i.e., the environment, the others agents, the communications, the external demands and the observations).

The knowledge and the belief set are generally incomplete, contradictory or error sensitive, being desirable to use semantics, procedures and implementations for dealing with the incomplete, contradictory, imperfect, wrong, nebulous or missing information problems, as well as the abduction, belief revision and argumentation operations. The Extended Logic Programming will be a flexible, rigorous and solid tool to contribute to obtain a solution for the problems just referred [9].

III. REPUTATION

Reputation is a concept or an opinion about a person or a virtual entity (e.g., an agent), with respect to social or professional properties, such as character, name, good fame, renown, credit or respectability. People can have evil repute or enjoy a good reputation. Otherwise, knowledge can be missing or incomplete in order to evaluate reputation, using quantitative or qualitative measures. Reputation may be also evaluated dynamically (e.g., overtime with respect to time points or time intervals). Reputation can be won or lost, and be evaluated from different sources, but the more relevant is the information obtained from direct contact and coming from a third peer [10] [11]:

- **Individual dimension** models the direct interaction between two agents;
- **Social Dimension:**
  - **Witness reputation** models the reputation about the target agent coming from other agents.
  - **Neighbourhood reputation** uses the social environments of the target agents; i.e. uses the values of reputation of the neighbour-hood with respect to the target agents and their relation with it; and
  - **System reputation** is the default value of reputation based on the role played by the target agents in the environments.

Reputation may be one important parameter to be taken into account in the process of trust evaluation in a negotiation, once it can be analyzed either in terms of the quality of knowledge it carries into the negotiation process, or the referentials it may impose.

IV. TRADITIONAL APPROACHES TO AUTOMATED NEGOTIATION

Negotiation is a mechanism where two or more parts, through a process of mutual concessions, try to reach a state of according. There are many ways to catalogue existing approaches to automated negotiations.

A. Game-theory

It is a branch of economics that studies the strategic interactions between self-interested economic entities. The researchers attempt to determine the optimal strat-
ergy by analyzing the interaction as a game among identical participants, and looking for its stability and equilibrium [12].

These approaches have some significant limitations from the computational point of view. They assume that agents have limitless computational resources and that the space of outcomes is completely known. In realistic environments these assumptions fail due to the limited processing and communication capabilities of the information systems [13].

B. Heuristic Approaches

Heuristics are rules that provide good and sufficient outcomes, and are frequently produced in contexts where the assumptions about agents’ rationality and resource are relaxed. Generally, they are based on empirical testing and evaluation.

In [14] various heuristic decision functions are used to evaluate and generate offers or proposals in a context of multi-attribute negotiation. These kinds of systems often lead to outcomes that are sub-optimal, once they do not check the complete space of possible outcomes and, on the other hand, it is difficult to predict how the system and the agent will proceed.

C. Argumentation

The approaches mentioned to above have more limitations [15]:
- Agents are not authorized to exchange any additional information, i.e., they restrict themselves to the proposals on the table;
- The agents’ utilities or preferences are assumed to be completely characterised in advance; i.e., they are not affected by the course of the interaction;
- They assume that agents’ utilities or preferences are fixed; i.e., an agent cannot influence its peers.

Traditional automated negotiation mechanisms do not improve the exchange of information. In the context of negotiation, argumentation is viewed as a mechanism to make possible the information exchange. Argument is viewed as a piece of information that may allow an agent to [16]:
- Justify its negotiation decision or option; and
- Influence others agents about the quality of its proposals.

The agent tries to turn its proposals more attractive, supplying additional information in the form of arguments.

But, in real negotiation situations, agents have not all the information and needs to reason. An agent must be able to construct its own arguments [13].

D. Logic-based Argumentation

The use of logic is welcome in the field of argumentation. Logic-based argumentation still presents a set of characteristics which can not be measured by a simplistic computational efficiency metric [8]:
- Adequacy to logic-based approaches to pre-argument reasoning: some agent development strategies define a stage that precedes the instant an agent starts to articulate an argument. This stage is called pre-argument reasoning and enables the agent to reason about such things as the right to deal some product or the right to deal with some counterpart. Due to the fundamental use of logic as a formalization tool and the manipulation of a logic Knowledge Base (KB), a set of rules is available in order to an argument be formulated;

- Similarity to the human reasoning processes: the use of logical mechanisms in reasoning enable easy construction of rules even by non-experts. On the other hand, the set of available rules (in an agent’s KB) is largely human-readable;
- Reasoning with incomplete information: the use of null values, in combination with negation by failure, enables the use of incomplete information and a reasoning mechanism that deals with uncertainty. An agent is able to construct arguments where some information is neither true nor false;

- Argument composition and extension: the set of logical elements (rules) which compose an argument may be extended in order to strengthen the argument conclusion, therefore innumerous compositions may be available, which allows for an easy adaptation to any specific kind of argument and/or problem (e.g., information exchange). On the other hand, taking an argument for A and the insertion of a rule such as B→A, an argument for B is trivially reached.

When a set of agents meets under an e-Commerce place, some kind of interaction may take place, namely by a process of offers and counter-offers, to support the modelling, analysis and enactment of the business process. The soundness of the process arises from the set of facts taken into consideration to produce an offer of counter-offer, i.e., the facts taken from an ordered logic theory, lead to a logical conclusion, organizing themselves into an argument. The importance of an argument has much to do with the time at which it arises; i.e., an argument may be deemed as a looser or a winner when facing a counter-argument, taking into account its sequence of evaluation. The exchange of offers and counter-offers must stop when some conditions are satisfied. These conditions may or may not lead to the definition of a winning set of arguments, which is the case in systems where the main concern goes to take full advantage of the argument evaluation.

V. THE AGENT KNOWLEDGE BASE

Knowledge representation techniques as a way to describe the real world, based on mechanical, logical or other means will be, always, a function of the systems ability to describe the existing world. Therefore, in the conception of a knowledge based system, it must be object of attention [17]:
- Existent Information: it may not be known in all its extension.
- Observed Information: it is perspectives by the experience, and obtained by contact or observation.
- Represented Information: with respect to a certain situation, it may be (ir)relevant to represent a given information. In spite of all the limitations, it is possible that observations made by different individuals, with distinct education and motivations, show the same set of fundamental data, function of its utility.

Prior to the characterization of the argument structure in terms of ELP productions, the agent knowledge base has to be addressed. It will be built around a set of logical terms subject to proof, then allowing for action justification and argument construction.

**Definition 1.** The knowledge available in each agent’s KB is made of logic clauses of the form $r_k \leftarrow p_i, p_j$, where $i, j, k \in N_0$, $p_i, p_j$ are literals; i.e., formula of the form $p \lor \neg p$, where $p$ is an atom, and where $r_k, \text{not}, p_{i+j}$, and $p_{i+1}, \text{not} p_{i+j}$ stand, respectively, for the clause’s identifier, the negation-by-failure operator, the rule’s consequent, and the rule’s antecedent. If $i=j=0$ the clause is called a fact and is represented as $r_k : p_i$.

An ELP program ($\Pi_{ELP}$) is seen as a set of clauses, as given by Definition 1. Therefore, the agent KB is taken from an ordered theory $OT=(T,>,S,$ and $\prec$ stand, respectively, for an agent’s knowledge base in clausal form, a non-circular ordering relation over such clauses, a set of priority rules, and a non-circular ordering relation over such rules. An argument (i.e., a proof, or series of reasons in support or refutation of a proposition) or arguments have their genesis on mental states seen as a consequence of the proof processes that go on unceasingly at the agent’s own knowledge about its states of awareness, consciousness or erudition. On the other hand, the mental states that have been referred to above are by themselves a product of reasoning processes over incomplete or unknown information; an argument may not only be evaluated in terms of true or false, but it may be quantified over the interval $[0,1]$ (e.g., an agent may be able to deal a product with one of its peers using a set of conditions $C_1$, however it is not known if it can do the same thing with a set of conditions $C_2$, which may lead to further confrontation).

This work is supported by the developments in [17] where the representation of incomplete information and the reasoning based on partial assumptions is studied, using the representation of null values [7] [18] to characterize abnormal or exceptional situations.

**A. Null Values**

The identification of null values emerges as a strategy for the enumeration of cases, whenever one tends to distinguish between situations where the answers are known (true or false) or unknown [17] [19].

The representation of null values will be scoped by the ELP. In this work, it will be considered two types of null values: the former will cater for the representation of unknown values, not necessarily from a given set of values, and the later will denote unknown values, from a given set of possible values.

Let us consider the predicate reputation(), which stands for an agent reputation and its value,

\[
\text{reputation: } \text{Entities} \times \text{Value}
\]

where the first argument denotes the agent and the later its reputation (e.g., reputation (Paul, 100) denotes that the reputation of the agent Paul has the value 100).

**Program 1:** Extension of the predicate that describes the reputation of agent Paul

In Program 1, the symbol $\neg$ stands for strong negation, denoting what should be interpreted as false, and the term not designates negation by failure or weak negation.

1) **Unknown**

Following the example given by Program 1, one may now admit that the reputation of the agent Cesar was not yet established. This situation will be represented by a null value, of the type unknown, that allows one to conclude that Cesar has a certain reputation, but it is not possible to be concise with respect to its value (Program 2).

**Program 2:** Extension of the predicate that sets the reputation of agent Cesar

The symbol $\bot$ denotes a null value of an undefined type, in the sense that it is assumed that any solution of the problem may be subscribed, but nothing is said about which solution one is speaking about. Computationally, it is not possible to determine, considering the positive information, the reputation of the agent Cesar; however, if one looks to the exceptions of the extension of predicate reputation() (fourth clause from Program 2, that sets the closure of predicate reputation()), it is discarded the possibility of any non-standard question to be assumed as false, when set with respect to the reputation of the agent Cesar.
2) **Unknown but Enumerated**

Consider now the example where the reputation of an agent, the agent José, is foreseen at 75, with a margin of error of 10%. It is not possible to be conclusive regarding the reputation as 75, or as 70 or even at 82.5 (Program 3). However, it is false that José’s reputation is 70 or 100. This example suggests that the lack of knowledge may be described by an enumerated set of possible values.

```
reputation( paul,100 )
reputation( cesar, 1 )
¬reputation( E, V ) ← not reputation( E, V ),
not exception(reputation( E, V ))
exception(reputation( E, V )) ← reputation( E, 1 )
exception( reputation( jose, V ) ) ← V ≥ 67.5 ∧ V ≤ 82.5
```

Program 3: Extension of the predicate that sets the reputation of the agent José

3) **Interpretation of Null Values**

The use of the body of knowledge that makes the agent’s knowledge base, set on the base of the formalisms referred to above, will be pursued in terms of the predicate demo(), using ELP as the logic programming language. Given a question, it returns a solution based on a set of assumptions. This meta-predicate will be defined as:

```
demo: Question × Answer
```

where Question denotes a theorem to be proved and Answer denotes a truth value: True (T), False (F) or Unknown (U) (Program 4).

```
demo( Q, T ) ← Q
demo( Q, F ) ← ¬Q
demo( Q, U ) ← not Q ∧ not ¬Q
¬demo( Q, V ) ← not demo( Q, V ), not exception( demo( Q, V ) )
```

Program 4: Extension of the meta-predicate demo()

The first clause of Program 4 is used to prove a question using the knowledge base positive information; the second clause is used to reject a question using the knowledge base negative information. The third and fourth clauses stand for themselves.

VI. THE QUALITY OF TRUST RECOGNITION

Based on the assumptions presented before, it is possible to establish mechanisms to analyze and process the information available in a way that turns feasible the study of the behaviour of virtual entities, in terms of its personification [18][20]. Situations involving forgetfulness, remembrance, learning or trust can be analyzed in the way proposed in this work; i.e., through the description of abnormal situations, declared as exceptions to a predicate extension.

A. **Characterization of a Problem**

Consider the example that follows, that intends to illustrate, through a practical application, the main contributions of this work.

```
reputation( paul, 1 ) :: 1.
exception(reputation( Agent, Value )::T) ← reputation( A, 1)::T.
¬reputation( paul, 100 ) :: 2.
exception(reputation( paul, 50 )::3).
exception(reputation( paul, 100 )::3).
exception(reputation( paul, 75 )::4).
exception(reputation( paul, 150 )::4).
```

Program 5: Extended logic programming excerpt that denotes the construction of Paul’s reputation.

In Program 5, at time point 1, there is an axiom stating that the system only knows that Paul has a reputation, but cannot be conclusive (it is not possible to evaluate the Paul’s reputation because there is no information at all). The second clause of Program 5 enforces that it must be considered false all the questions with lack of information and that can not be taken as exceptions to the extension of predicate reputation().

At time point 2, there is a proposition that the Paul’s reputation is 100.

At time point 3, it is known that the Paul’s reputation is either 50 or 100.

At time point 4, the Paul’s reputation is either 75 or 150.

At time point 5, the reputation of Paul is 200.

Consequently, in terms of the temporal axis $t_1 \rightarrow t_2 \rightarrow t_3 \rightarrow t_4$, an analysis of these values leads one to the conclusion that the knowledge base learned something (i.e., it acknowledge the Paul’s reputation).

B. **The System Semantics**

It is intended to evaluate the quality of the information used in a dialog, being it a simple chat or a commercial transaction, by measuring the trustfulness of our peers or opponents. For example, in Program 5, the situation represented by reputation(paul,1)::1 corresponds to a case where the quality of the information is given in the form:

$$G_{reputation \{ paul \}, \forall \_1} = \lim \frac{1}{N} = 0$$

where N denotes the cardinality of the set of clauses in the form exception(reputation(paul,_)::1), i.e., the clauses that make the exceptions to the extension of the predicate reputation(). The symbol “_” stands for any value that may be endorsed as Paul’s reputation. In the situation given by reputation(paul, 100)::2, it corresponds to a case where the quality of the information is given in the form:
i.e., the cardinality of the set of the exceptions to the extension of predicate \( \text{reputation}() \) decreased; it means that the system is becoming aware of a possible change of course, with respect to possible moves concerning Paul’s reputation.

Accordingly, in instant 4 (four), one has:

\[
\text{G} \left( \text{reputation} (\text{paul}, V), 4 \right) = \frac{1}{4} = 0.25
\]

VII. STRATEGY

The strategy for developing a consistent and sound approach to the use of agents in e-Commerce environments is based on a constructive view of the universe of discourse, and the agent attitudes [21], in terms of:

- Architecture development: to determine and analyze each feature an agent should have to deal with a particular area of interest; to design the flow of information across the different agent building blocks;
- Process quantification: to quantify each metric and sub-process with which the agents have to deal with (e.g., trust);
- Reasoning procedures: each agent is in need of a set of axioms that may serve as the main guidelines for the negotiation processes, i.e., set of factors that are to be taken into account or evaluated, before any kind of opposition among agents may take place; and
- Process formalization: argumentation is in need of a formal treatment in order to be consistent and for the agents to act or react in a reasonable way.

The use of logic in the specification of argument-based negotiation aims not only at the definition of the best dealing strategies, but also to mimic the inference mechanisms that take place in the mind of the user, in order to enable a true feasible implementation [22].

Automatic argument generation relies on three types of proceedings:

- Proto-autonomy: at each round the user redefines the global course of action (e.g., attack/denial) and only then the incumbent agent indicates which arguments (first in time, order or rank) are to be launched;
- Semi-autonomy: to the system’s agents, it must be granted access to historical argument based databases, in order to allow the incumbent agents to use different data or knowledge handling techniques to extrapolate the intended course of action when a new argument is on the table; and
- Full-autonomy: using Machine Learning tools the incumbent agents may progressively learn and adapt their mental states to a particular sort of arguments which are exchanged.

The process of argument generation may be seen as a logic-guided mechanism that acts in order to support some utterance (e.g., it is possible for an human being to act in support of an argument, obtained on the basis of the formalism referred to above, when a knowledge base of local or global knowledge is made available [22]. Each term that is part of an argument may come from one of two main sources: global or proper knowledge. Global knowledge is shared by the intervening entities and is, therefore, independent of a particular experience. Proper knowledge derives from sources that are not common to every agent, giving way to the possibility of contradictory conclusions upon confrontation. The knowledge base embedded in each agent may have a quite different genesis with respect to other approaches. The use of global or proper knowledge is a restriction of the system capacity to determine the winner/loser of a confrontation. As expected, proper knowledge is not the best starting-point for a premise denial/attack (e.g., a claim such as my experience tells me I sold item X for Y euro is difficult to be stated as false by a counterpart, once he/she can not say which is the grasp of life of the opponent).

VIII. CONCLUSIONS

The main contributions of the present work may be understood in terms of the definition of a common ground to situate the agent reasoning mechanisms in e-Commerce environments; the use of formal tools (e.g., extended logic programming) to describe the rational behaviour of the entities involved (e.g., in terms of trust and reputation); the use of incomplete information and previous experiences in the reasoning process; the bridging between legal argumentation and argument-based negotiation; and the establishment of sound syntactic and semantic tools for logic-based argumentation.

ELP proved to be a well adequate tool for knowledge representation and reasoning, in particular when one intend to endorse situations where the information
is vague or incomplete, which is the case when there is the intention to represent at the agent level properties and attitudes only found in the human beings.

In e-Commerce the introduction of agent-based technology makes possible the use of a set of high-level reasoning tools, leading to architectures that base themselves on the assumption that in a real-world environment entities act as mediators. Under this umbrella, simple inference and knowledge acquisition mechanisms are not sufficient to provide a sustainable and credible system. Traditional computational models need to be complemented with procedures native to the business. Indeed, the issues posed by e-Commerce need to be addressed in a formal fashion. E-Commerce is an area that poses particular problems to the use of agent-based software. Although applications in this area are particularly suited to be solved by agent based systems, no formal development process has yet been devised for such field of expertise. In this work, the processes among agents are set and quantified, and the reasoning mechanisms are recognized. The processes involved in e-Commerce which are difficult to incorporate into traditional systems, revolve around subjective business parameters. Indeed, parameters such as gratitude, agreement and trust among parties are nonlinearities, which need to be considered in order to develop a feasible e-Commerce system. This information is to be taken into account when drawing up a strategic plan of action. However, once subjective parameters have been quantified, some reasoning must take place before any argument is exchanged with potential counterparts. This stage, which has been called pre-negotiation reasoning deals with the existence of incomplete information and delineates logical conclusions upon an agent knowledge base (e.g., is it agent A able to deal product P with agent B at time T). Exchanging justified information provides everyone with enough knowledge to try a common understanding with its counterparts in time.

REFERENCES

