Rule-Based Automated Price Negotiation: Overview and Experiment

Costin Bădică[1], Maria Ganzha[2], and Marcin Paprzycki[3]

[1] University of Craiova, Software Engineering Department
Bvd.Decebal 107, Craiova, RO-200440, Romania
badica_costin@software.ucv.ro
[2] Elbląg University of Humanities and Economy, ul Lotnicza 2, 82-300 Elbląg, Poland
ganzha@euh-e.edu.pl
[3] Computer Science Institute, Warsaw School of Social Psychology, 03-815 Warsaw, Poland
Marcin.Paprzycki@swps.edu.pl

Abstract. The idea of automating e-commerce transactions attracted a lot of interest during the last years. Multi-agent systems are claimed to be one of promising software technologies for achieving this goal. In this paper we summarize state-of-the-art in rule-based approaches to automated negotiations and present initial experimental results with our own implementation of a rule-based price negotiation mechanism in a model e-commerce multi-agent system. The experimental scenario considers multiple buyer agents involved in multiple English auctions that are performed in parallel.

1 Introduction

During last years, interest in e-commerce has shifted from simple Web presence of a business to advanced use of e-commerce technologies in order to support growth of business itself — by improving its efficacy and profitability. Therefore, the idea of automating e-commerce transactions attracted a lot of research interest ([15]).

Most of currently existing e-commerce systems involve humans that make most important decisions in various activities along the lifeline of an e-commerce transaction. At the same time, software agents are claimed to be one of the best technologies for automating e-commerce processes. It is expected that intelligent agents will be able to substantially reduce (if not eliminate) need for human involvement — in all but most crucial decisions. In this context, we have set up a project to contribute development of such an agent system [11]. In particular our project has two main goals: (1) to build a large-scale implementation approximating an e-commerce environment; (2) to develop a tool that we will be able to use for modeling various e-commerce scenarios.

E-commerce research proposes that when digital technologies are utilized to mediate commercial transactions, then the complete process can be conceptualized as consisting of four phases: (i) *pre-contractual phase* including activities like need identification, product brokering, merchant brokering, and matchmaking; (ii) *negotiation* where negotiation participants negotiate according to the rules of a particular market mechanism and using their private negotiation strategies; (iii) *contract execution* including activities like: order submission, logistics, and payment; and (iv) *post-contractual phase*

L. Rutkowski et al. (Eds.): ICAISC 2006, LNAI 4029, pp. 1050–1059, 2006.

that includes activities like collecting managerial information and product or service evaluation.

Focus of this paper is on the *negotiation phase* of a transaction taking place in a multi-agent e-commerce system that we have started to develop ([11]). As a part of this work we are interested in endowing our agents with flexibility necessary to engage in unknown in advance forms of negotiations using rule-based approaches. We start our presentation with an overview of state-of-the-art of rule representations in automated negotiations. We follow with a brief summary of the architecture of our proposed system that uses a rule-based framework for enforcing specific negotiation mechanisms, together with a sample scenario. Finally, we discuss some experimental results of our implementation.

2 Background on Rule-Based Negotiation

Rules have been indicated as a very promising technique for formalizing multi-agent negotiations ([4,7,8,10,16,18,22,27,28]). When considering design of systems for automated negotiations it is typically the case that *negotiation protocols* (or *mechanisms*) that define "rules of encounter" between participants and *negotiation strategies* that define behaviors aiming at achieving a desired outcome are distinguished. However, rule representations were proposed for both negotiation mechanisms ([7,24,16,18,27,28]) and strategies ([10,22,8]). Let us now summarize most important developments in the area of rule based approaches to automated negotiations.

In our work we follow a rule-based framework for enforcing specific negotiation mechanisms inspired by [7]. Its authors sketched a complete framework for implementing portable agent negotiations that comprises: (1) negotiation infrastructure, (2) generic negotiation protocol and (3) taxonomy of declarative rules. The *negotiation infrastructure* defines roles of negotiation participants and of a host. Participants exchange proposals within a negotiation locale managed by the host. The *generic negotiation protocol* defines three phases of a negotiation: admission, exchange of proposals and formation of an agreement, in terms of how and when messages should be exchanged between the host and participants. *Negotiation rules* are used for enforcing the negotiation mechanism. Rules are organized into a taxonomy: rules for participants admission to negotiations, rules for checking the validity of proposals, rules for protocol enforcement, rules for updating the negotiation status and informing participants, rules for agreement formation and rules for controlling the negotiation termination.

The proposal for formalizing negotiations introduced in [24] goes beyond the framework of [7]. Its authors suggest to use an ontology for expressing negotiation protocols. Whenever an agent is admitted to negotiation it also obtains a specification of the negotiation rules in terms of the shared ontology. In some sense, the negotiation template used by our implementation (see [4]) is a "simplified" negotiation ontology and the participants must be able to "understand" the slots defined in the template. This approach is exemplified with a sample scenario. The ontology approach introduced in [24] is taken further in [23] by investigating how the ontology can be used to tune the negotiation strategy of participant agents. However, paper [24] contains neither implementation details, nor experimental results. Furthermore, we were not able to obtain a complete version of the ontology described in the paper.

In [27,28] a mathematical characterization of auction rules for parameterizing the auction design space is introduced. The proposed parametrization is organized along three axes: i) *bidding rules* – state when bids may be posted, updated or withdrawn; ii) *clearing policy* – states how the auction commands resource allocation (including auctioned items and money) between auction participants (this corresponds roughly to agreement making in our approach); iii) *information revelation policy* – states how and what intermediate auction information is supplied to participating agents.

In [18] authors developed a special declarative language CLP ("Courteous Logic Programs as KR") for expressing and reasoning about contracts and negotiations. This project was a continuation of the Michigan AuctionBot project ([26]), and its authors focused on the automatic configuration of negotiations based on a contract and showed how rules generated during the negotiation process can be combined with the partial contract to form an executable final contract. Background knowledge supporting this infrastructure was embodied in three CLP rule sets: *Auction-Configuration*, *Auction-Space*, and *Auctionbot-Mapping*. *Auction-Configuration* supports reasoning about alternative negotiation structures and also specifies how to split contract into an array of auctions. *Auction-Space* implements a cleaner, more general parameterization of the auction design space, imposes constraints and conditional defaults on parameters, and infers auction parameters from higher-level knowledge about the negotiation. *AuctionBot-Mapping* maps the *Auction-Space* parameterization to the existing set of *AuctionBot* parameters. Unfortunately, we were not able to find any information about continuation of this interesting project.

In [16] an implementation of a new rule-based scripting language (*AB3D*) for expressing auction mechanisms is reported. The design and implementation of AB3D were primarily influenced by the parametrization of the auction design space defined in [27,28] and the previous experiences with the Michigan Internet AuctionBot ([26]). According to [16], A3BD allows the initialization of auction parameters, the definition of rules for triggering auction events, the declaration of user variables and the definition of rules for controlling bid admissibility.

A formal executable approach for defining the strategy of agents participating in negotiations using defeasible logic programs is reported in [12] and [10]. The approach is demonstrated on English auctions and bargaining with multiple parties by indicating sets of rules for describing strategies of participating agents. However, paper [12] contains neither implementation details, nor experimental results.

In [22] a preliminary implementation of a system of agents that negotiate using strategies expressed in defeasible logic is described. The implementation is demonstrated with a bargaining scenario involving one buyer and one seller agent. The buyer strategy is defined by a defeasible logic program. Note that the implementation reported in [22] builds on the architecture of negotiating agents previously introduced in [10]. Note also that defeasible logic programs are able to express courteous logic programs proposed in [18] and yet to support efficient reasoning, which suggest that they might be the appropriate representation formalism of negotiation strategies.

The CONSENSUS system that enables agents to engage in combined negotiations was presented in [8]. CONSENSUS allows agents to negotiate different complementary items on separate servers on behalf of human users. Each CONSENSUS agent uses a rule base partitioned into: i) *basic rules* that determine the negotiation protocol, ii) *strategy rules* that determine the negotiation strategy and iii) *coordination rules* that determine the knowledge for assuring that either all of the complementary items or none are purchased. Note that in CONSENSUS the rule-based approach is taken beyond mechanism and strategy representation to capture also coordination knowledge.

Another interesting work is the open environment for automated negotiations specifically targeted to auctions – auction reference model (ARM, [20]) and its associated declarative auction specification language (DAL, [19]). It should be noted that, while not explicitly using rules, a DAL specification actually models the flow of an auction using a rule-based approach. DAL constructs comprise the following: views, validations, transitions and agreement generators. Views are analogous to visibility rules, validations are analogous to validity and protocol enforcement rules, transitions are analogous to update rules and agreement generators are analogous to agreement formation and negotiation life-cycle rules.

Before proceeding further, let us make the following remark. E-commerce is seen as one of the key services of modern information society and therefore, the ability of software agents to discover remote markets and engage in commercial transactions governed by market mechanisms unknown in advance, is of primary importance. Rules constitute a very promising approach to describing negotiation processes (see for instance all references cited in this section). However, a key aspect for success of automated negotiations, that already generated some interest in the research community, is the development of a truly open rule-based semantic description of the market mechanism [7,4,19,20,24,23]. As our research indicates, we are still quite far from that vision of software agents needing only minimal pre-compiled knowledge to enable them to ”sense” the negotiation mechanism and ”tune” the negotiation strategy accordingly. It is exactly this issue that catalyzes our work and that differentiates it from previous works, making us to proceed further.

3 System Description and Experiment

3.1 Conceptual Architecture

Our system acts as a distributed marketplace in which agents perform functions typically observed in e-commerce ([11]). E-shops are represented by shop and seller agents, while e-buyers are represented by client and buyer agents. In Figure 1 we present Use-Case diagram of the complete system. Outside o bounds of the system we can see *User-Client* who will attempt at buying products from one of the e-shops and *User-Seller* who tries to sell products in her e-store. Let us now briefly summarize the most important agents appearing in the system and their functionalities (for a complete discussion of the system see [3,5,6]). *User-Client* is represented by the *Client Agent* (*CA*). The *CA* is completely autonomous and as soon as the decision to purchase product *P* is communicated by the *User-Client*, it will work until either the product is purchased or, due to the market circumstances (e.g. prices are to high) purchase is abandoned.

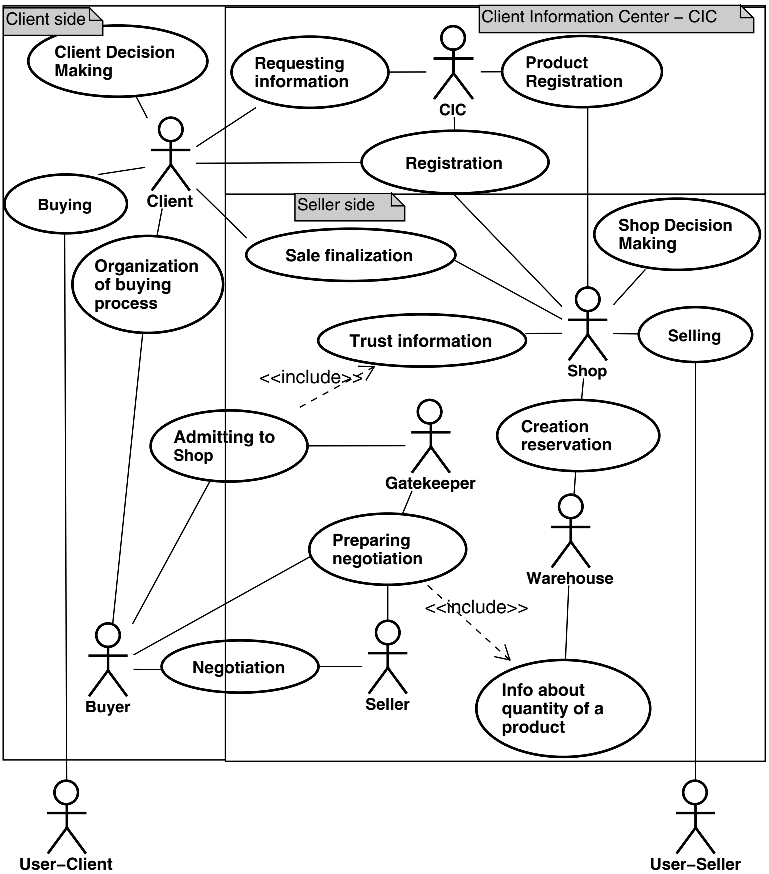

Fig. 1. Use Case diagram of the proposed agent-based e-commerce system

The *CA* communicates with the *Client Information Center* (*CIC*) agent which contains complete information which e-stores sell which products. For each store that sells the requested product, the *CA* delegates a single *Buyer Agent* (*BA*) with a mission to be involved in price negotiations and if successful, possibly attempt at making a purchase (successful price negotiations result in a product reservation for a specific time period; after which products that have not been actually purchased are returned to the pool of products available for sale). Since multiple *BA*s representing the same *CA* can win price negotiations and report to the *CA*, it is the *CA* that makes the decision if either of available offers is good enough to make a purchase. *Buyer Agents* either migrate to the negotiation host or are created locally [6]. They can participate in negotiations only if the *Gatekeeper Agent* (*GA*) allows this. The *GA* utilizes *trust information* to evaluate if a given *BA* should be admitted (*BA*s that win price negotiations but do not make a purchase may be barred from subsequent negotiations). The *GA* is one of agents that represent that e-store and is created by the *Shop Agent* (*SA*). The *SA* is the central manager of the e-shop. Facilitating the selling process, the *SA* utilizes the (*GA*), as well as a *Warehouse Agent* (*WA*) that is responsible for inventory and reservation management; and a set of *Seller Agents* (*SeA*) that negotiate price with incoming *BA*s.

In our experiments we considered simplified version of this scenario that involves a single *Shop Agent* and n *Client Agent*s CA_i, $1 \leq i \leq n$. The *SA* is selling m products $\mathcal{P} = \{1, 2, \ldots, m\}$. We assume that each client agent CA_i, $1 \leq i \leq n$, is seeking a set $\mathcal{P}_i \subseteq \mathcal{P}$ of products (we therefore restrict our attention to the case where all sought products are available through the SA). The SA is using m *Seller Agents* SeA_j, $1 \leq j \leq m$ and each SeA_j is responsible for selling a single product j. Each CA_i is using buyer agents BA_{ik} to purchase products from the set $\mathcal{P}_i$. Each BA_{ik} is responsible for negotiating and buying exactly one product $k \in \mathcal{P}_i$, $1 \leq i \leq n$. To attempt purchase *Buyer Agent*s BA_{ik} migrate to the SA and engage in negotiations; a BA_{ik}, that was spawned by the *Client Agent* CA_i, will engage in negotiation with seller SeA_k, to purchase product k. This simple scenario is sufficient for the purpose of our paper, i.e. to illustrate our rule-based system and show how a number of rule-based automated negotiations can be performed concurrently. In this setting, each *Seller Agent* SeA_j plays the role of a negotiation host defined in [7]. Therefore, in our system, we have exactly m instances of the framework described in [7]. Each instance is managing a separate "negotiation locale", while all instances are linked to the *Shop Agent*. For each instance we have one separate set of rules together with a negotiation template that describes the negotiation mechanism implemented by that host. Note that each seller may use a different negotiation mechanism (different form of an auction, or an auction characterized by different parameters, such as the starting price or the bidding increment). See [4] for the details of our implementation of this conceptual architecture using JADE ([13]) and JESS ([14]).

3.2 Rule-Based Representation of English Auctions

For the purpose of this paper we have set up our system for a particular negotiation scenario involving English auctions. Technically, English auctions are single-item, first-price, open-cry, ascending auctions ([15],[25]). In an English auction there is a single item (or a collection of products treated as a single item) sold by a single seller and many buyers bidding against each other for buying that item. Usually, there is a time limit for ending the auction, a seller reservation price that must be met by the winning bid for the item to be sold and a minimum value of the bid increment. A new bid must be higher than the currently highest bid plus a minimal bid increment in order to be accepted. All the bids are visible to all the auction participants.

The constraints describing English auctions were encoded as a modularized set of JESS rules. The rules were then used to initialize rule inference engines encapsulated by the negotiation hosts [4]).

Let us now consider a few sample rules for representing English auctions. These rules are described informally using a pseudo-code notation that is independent of any implementation-level language (like JESS).

POSTING-BUYER rule specifies that a $buyer$ participant can post a proposal whenever there is an offer already posted by a $seller$ participant.

POSTING-BUYER
IF
There is a valid proposal Pr of a participant with role $buyer$ $\wedge$
There is an active proposal of a participant with role $seller$
THEN
Proposal Pr is posted

IMPROVEMENT-BUYER rule specifies that a $buyer$ participant must post a proposal with a price that must overbid the currently highest id with at least a given increment)that is a parameter of the auction).

IMPROVEMENT-BUYER
IF
Negotiation is on goods A $\wedge$
Bid increment is Inc $\wedge$
Currently highest bid is B $\wedge$
Proposal Pr on goods A with price P was posted by a $buyer$ $\wedge$
$P > B + Inc$
THEN
Proposal Pr is active

AGREEMENT-FORMATION rule specifies that whenever agreement formation is triggered, if the currently highest bid is greater than the $seller$ reservation price (that it is not disclosed to the participants), an agreement is formed between the submitter of the highest bid and the $seller$.

AGREEMENT-FORMATION
IF
The currently highest bid is B and was submitted by $buyer$ $S1$ $\wedge$
There is an active proposal of $seller$ $S2$ with price P $\wedge$
Negotiation is on goods A $\wedge$
$B \geq P$
THEN
An agreement of $S1$ with $S2$ to transact goods A at price $P1$ is formed

3.3 Participants Strategy

Strategies of participant agents are defined in accordance with the negotiation protocol (i.e. English auctions in this particular setting). Basically, the strategy defines if and when a participant will submit a proposal depending on what are the values of its parameters. For the time being we opted for a simple solution: the participant submits a first bid immediately after it was granted admission and whenever it gets a notification that another participant issued an accepted proposal. The value of the bid is equal the sum of the currently highest bid and an increment value that is private to the participant. Each participant has its own valuation of the negotiated product. If the value of the new bid exceeds this value then the proposal submission is canceled (given product became "too expensive" for a given *BA*). Note that in the case of an English auction there is no particular strategy for the *Seller Agent* as it plays only a passive role.

Agent strategies were implemented in Java as participant agent behaviors ([13]). In the future we plan to design the system in such a way that strategies will also be represented in the rule-based form ([10,12]). This will provide us with the required flexibility to easily add multiple strategies to our implementation. Obviously, in practice, this form of strategy representation is required only for more involved forms of price negotiations (where utilization of complicated strategies makes much more sense).

3.4 Experiment

In the experiment we considered $m = 10$ products and $n = 12$ clients seeking all of them, i.e. $\mathcal{P}_i = \mathcal{P}$ for all $1 \leq i \leq 10$. The auction parameters were the same for all auctions: reservation price 50 and minimum bid increment 5. Clients reservation prices were randomly selected from the interval [50,72] and their bid increments were randomly selected from the interval [7,17].

In this experiment 143 agents were created: 1 shop SA, 10 sellers SeA_i, $1 \leq i \leq 10$, 12 clients CA_i, $1 \leq i \leq 12$, and 120 buyers BA_{ik}, $1 \leq i \leq 12$, $1 \leq k \leq 10$, and 10 English auctions were run concurrently. One separate JESS rule engine was also created for each English auction (therefore a total of 10 JESS rule engines were run in parallel). The average number of messages exchanged per negotiation was approximately 100 and all the auctions finished successfully. This means that a total of more than 1000 messages was exchanged during negotiations. While the total number of agents and messages is still small (for instance in comparison with these reported in [9], this experiments indicates that the proposed approach has good potential for supporting experiments on large-scale.

Figure 2 shows messages exchanged between the seller SeA_1 and buyers BA_{i1}, $1 \leq i \leq 12$ that were captured with the help of the JADE sniffer agent.

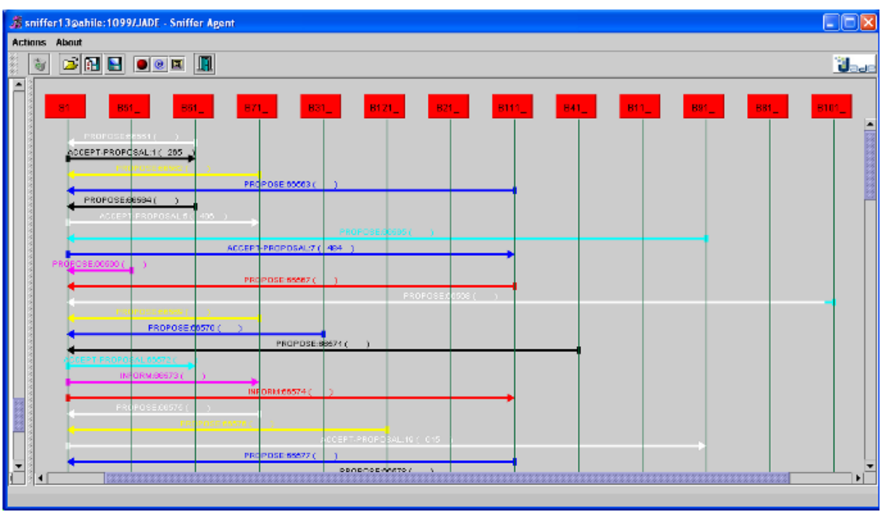

Fig. 2. Negotiation of a seller with 12 buyers in an English auction

4 Conclusions and Future Work

In this paper we discussed rule-based approaches for automated negotiation in a model multi-agent e-commerce systems. Our discussion was supplemented by providing experimental results obtained using our own implementation of a rule-based price automated negotiation framework. The results support the claim that rules are a feasible and scalable technology for approaching flexible automated negotiation in e-commerce.

As future work we plan to: (i) complete the integration of the rule-based framework into our agent-based model e-commerce system; (ii) to asses the generality of our implementation by extending it to include other price negotiation mechanisms; (iii) to conceptualize representation and ways to efficiently implement multiple strategy modules; iv) to investigate the applicability of rule-markup languages ([21]) for devising an open rule-representation of negotiation mechanisms. We will report on our progress in subsequent papers.

References

1. Bădică, C., Ganzha, M., Paprzycki, M., Pîrvănescu, A.: Combining Rule-Based and Plug-in Components in Agents for Flexible Dynamic Negotiations. In: M. Pěchouček, P. Petta, L.Z. Varga (eds.): *Proceedings of CEEMAS'05*, Budapest, Hungary. LNAI 3690, Springer Verlag (2005) 555–558.
2. Bădică, C., Ganzha, M., Paprzycki, M., Pîrvănescu, A.: Experimenting With a Multi-Agent E-Commerce Environment. In: *Proceedings of PaCT'2005*, Krasnoyarsk, Russia. LNCS 3606 Springer Verlag, (2005) 393–401.
3. Bădică, C., Ganzha, M., Paprzycki, M.: Mobile Agents in a Multi-Agent E-Commerce System. In: *Proceedings of the SYNASC 2005 Conference*, Timişoara, Romania. IEEE Computer Society Press, Los Alamitos, CA, (2005) 207–214.
4. Bădică, C., Bădiţă, A., Ganzha, M., Iordache, A., Paprzycki, M.: Rule-Based Framework for Automated Negotiation: Initial Implementation. In: A. Adi, S. Stoutenburg, S. Tabet (eds.): *Proceedings RuleML'2005*, Galway, Ireland. LNCS 3791, Springer Verlag (2005) 193–198.
5. Bădică, C., Ganzha, M., Paprzycki, M.: UML Models of Agents in a Multi-Agent E-Commerce System. In: *Proc. ICEBE'2005*, Beijing, China. IEEE Computer Society Press, Los Alamitos, CA, (2005) 56–61.
6. Bădică, C., Ganzha, M., Paprzycki, M.: Two Approaches to Code Mobility in an Agent-based E-commerce System. In: C. Ardil (ed.), *Enformatika*, Volume 7, (2005) 101–107.
7. Bartolini, C., Preist, C., Jennings, N.R.: A Software Framework for Automated Negotiation. In: *Proceedings of SELMAS'2004*, LNCS 3390, Springer Verlag (2005) 213–235.
8. Benyoucef, M., Alj, H., Levy, K., Keller, R.K.: A Rule-Driven Approach for Defining the Behaviour of Negotiating Software Agents. In: J.Plaice et al. (eds.): *Proceedings of DCW'2002*, LNCS 2468, Springer verlag (2002) 165–181.
9. Chmiel, K., Tomiak, D., Gawinecki, M., Karczmarek, P., Szymczak, Paprzycki, M.: Testing the Efficiency of JADE Agent Platform. In: *Proceedings of the* 3^{rd} *International Symposium on Parallel and Distributed Computing*, Cork, Ireland. IEEE Computer Society Press, Los Alamitos, CA, USA, (2004), 49–57.
10. Dumas, M., Governatori, G., ter Hofstede, A.H.M., Oaks, P.: A Formal Approach to Negotiating Agents Development. In: *Electronic Commerce Research and Applications*, Vol.1, Issue 2 Summer, Elsevier Science, (2002) 193–207.

11. Ganzha, M., Paprzycki, M., Pîrvănescu, A., Bădică, C., Abraham, A.: JADE-based Multi-Agent E-commerce Environment: Initial Implementation, In: *Analele Universităţii din Timişoara, Seria Matematică-Informatică* Vol. XLII, Fasc. special, (2004), 79–100.
12. Governatori, G., Dumas, M., ter Hofstede, A.H.M., and Oaks, P.: A formal approach to protocols and strategies for (legal) negotiation. In: Henry Prakken, editor, *Procedings of the 8th International Conference on Artificial Intelligence and Law*, IAAIL, ACM Press, (2001) 168–177.
13. JADE: Java Agent Development Framework. See `http://jade.cselt.it`.
14. JESS: Java Expert System Shell. See `http://herzberg.ca.sandia.gov/jess/`.
15. Laudon, K.C., Traver, C.G.: *E-commerce. business. technology. society* (2^{nd} ed.). Pearson Addison-Wesley, (2004).
16. Lochner, K.M., Wellman, M.P.: Rule-Based Specification of Auction Mechanisms. In: *Proc. AAMAS'04*, ACM Press, New York, USA, (2004).
17. Lomuscio, A.R., Wooldridge, M., Jennings, N.R.: A classification scheme for negotiation in electronic commerce. In: F. Dignum, C. Sierra (Eds.): *Agent Mediated Electronic Commerce: The European AgentLink Perspective*, LNCS 1991, Springer Verlag (2002) 19–33.
18. Reeves, D. M.; Grosof, B. N.; Wellman, M. P.; and Chan, H. Y.: *Toward a declarative language for negotiating executable contracts.* In: AAAI-99 Workshop on ArtiïňĄcial Intelligence in Electronic Commerce (AIEC-1999).
19. Rolli, D., Eberhart, A.: An Auction Reference Model for Describing and Running Auctions. In: *Electronic Markets – The International Journal* (2005).
20. Rolli, D., Luckner, S., Gimpel, A.: A Descriptive Auction Language. Electronic Markets. In: *7 Internationale Tagung Wirtschaftsinformatik*, Bamberg, Germany, (2005).
21. RuleML Initiative. See `http://www.ruleml.org`.
22. Skylogiannis, T., Antoniou, G., Bassiliades, N.: A System for Automated Agent Negotiation with Defeasible Logic-Based Strategies – Preliminary Report. In: Boley, H., Antoniou, G. (eds): *Proceedings RuleML'04*, Hiroshima, Japan. LNCS 3323 Springer-Verlag (2004) 205–213.
23. Tamma, V., Phelps, S., Dickinson, I., Wooldridge, M.: Ontologies for Supporting Negotiation in E-Commerce. In: *Engineering Applications of Artificial Intelligence*, 18, Elsevier (2005) 223–238.
24. Tamma, V., Wooldridge, M., Dickinson, I: An Ontology Based Approach to Automated Negotiation. In: *Proceedings AMEC'02: Agent Mediated Electronic Commerce*, LNAI 2531, Springer-Verlag (2002) 219–237.
25. Wooldridge, M.: *An Introduction to MultiAgent Systems*, John Wiley & Sons, (2002).
26. Wurman, P.R., Wellman, M.P., Walsh, W.E.: The Michigan Internet AcutionBot: A Configuarable Auction Server for Human and Software Agents. In: *Proceedings of the second international conference on Autonomous agents. Agents'98*, Minneapolis, USA. ACM Press, New York, USA, (1998), 301-308.
27. Wurman, P.R., Wellman, M.P., Walsh, W.E.: A Parameterization of the Auction Design Space. In: *Games and Economic Behavior*, 35, Vol. 1/2 (2001), 271–303.
28. Wurman, P.R., Wellman, M.P., Walsh, W.E.: Specifying Rules for Electronic Auctions. In: *AI Magazine* (2002), 23(3), 15–23.