An effective Negotiation Strategy for simple buyer response in E-commerce environment

Malamati Louta, *Member, IEEE,* Ioanna Roussaki, *Member, IEEE*, and Lambros Pechlivanos

Abstract— Electronic commerce is expected to dominate the market if coupled with the appropriate technologies and mechanisms. Mobile agents are one of the means that may enhance the intelligence and improve the efficiency of systems in the e-marketplace. In this paper, we propose a dynamic multilateral negotiation model and an efficient negotiation strategy that can be used to extend the functionality of autonomous agents, so that they reach to an agreement aiming to maximise their owner's utility. The framework considers contract and decision issues, is based on real market conditions, adopts a light ranking mechanism that does not require a complicated rationale on behalf of the buyer agents, and has been empirically evaluated.

***Index Terms*— Intelligent Agents, Negotiation Protocol & Model, Strategy, Ranking Mechanism, Simulated Annealing.**

I. INTRODUCTION

The last few years we have witnessed a rapid expansion of business carried out online, which constitutes electronic commerce a field dominating present and future transactions. While current e-commerce systems offer advantages to both consumers and merchants, it is often the case that they offer little more than electronic catalogues on which credit card payments can be arranged online. In order to harness its full potential and achieve the degree of automation required by e-commerce applications, a new technology is necessitated. Agent technology, which is already affecting almost every aspect of computing, seems to play a leading role, enabling a new, more flexible, generation of e-commerce systems. In such systems, automated software agents participate in trading activities on behalf of their owner. This paper is based upon the notion of interacting agents, which exhibit properties such as autonomy, reactivation, and pro-activation, in order to achieve particular objectives and accomplish the goals of their owners in a negotiation environment.

One of the major changes that will be brought about by e-commerce is that dynamic pricing and personalisation of offers will become the norm for many transactions requiring thus extended negotiation capabilities. Mobile intelligent agents can act as mediators in five of the six e-commerce phases [1]: need identification, product brokering, buyer coalition formation, merchant brokering and negotiation. This paper explores the role and behaviour of agents in the negotiation phase.

Negotiation may be defined as "*the process by which a joint decision is made by two or more parties. The parties first verbalise contradictory demands and then move towards agreement by a process of concession or search for new alternatives*" [2]. In human negotiations, the parties bargain to determine the price or other transaction terms. In an automated negotiation, software agents engage in broadly similar processes to achieve the same end. Automated negotiation is a very broad and encompassing field. For this reason, it is important to understand the dimensions and range of options that are available. When building autonomous agents capable of sophisticated and flexible negotiation, three broad areas need to be considered [3]: (i) what negotiation protocol and model will be adopted, (ii) what are the issues over which negotiation will take place, and (iii) what negotiation strategies will the agents employ. The negotiation protocol defines the "rules of encounter" between the agents [4]. Then, depending on the goals set for the agents and the negotiation protocol, the negotiation strategies are determined [5]. Given the wide variety of possibilities, there is no universally best approach or technique for automated negotiations [6], rather protocols and strategies need to be set according to the prevailing situation. Thus, a design repository for interactions in agent-mediated e-commerce will be produced, enabling design expertise and know-how to be shared between developers.

This paper concentrates predominantly on the first issue, proposing a negotiation protocol to be employed in an automatic multi-lateral multi-step negotiation model and on the third point providing an efficient negotiation strategy for the electronic *Business-to-Consumer* marketplace. In this framework, the roles of the negotiation agents may be classified into two main categories that, in principle, are in conflict. Thus, the negotiating agents may be divided into two subsets: $\{Agents\} = \{SellerAgents\} \cup \{BuyerAgents\}$. The *Buyer Agents* (*BAs*) and the *Seller Agents* (*SAs*) are considered to be self-interested, aiming to maximise their owners' profit, while the maximisation of the social welfare is not an objective in the overall design. The authors exploit a multi-step negotiation mechanism, which demonstrates inherent computational and communication advantages over single step mechanisms in such complex frameworks [7]. In essence, the negotiating agents hold private information, which may be revealed incrementally, only on an as-needed basis, while the proposed solution is efficient when the disclosure of information is not acceptable, possible, or desired. The negotiation environment studied covers multi-issue contracts and multiparty situations, while being a highly dynamic one, in the sense that its variables, attributes and leading objectives may change over time. The problem considered assumes that SAs

---

Malamati Louta and Ioanna Roussaki are with the National Technical University of Athens, School of Electrical and Computer Engineering, Athens, Greece (telephone: +30-210-7722422, e-mail: {louta,nanario}@telecom.ntua.gr).

Lambros Pechlivanos is with Athens University of Economics & Business, Department of International and European Economic Studies, Athens, Greece (telephone: +30-210-8203728, e-mail: lpech@aueb. gr).

and/or BAs face strict deadlines. In this framework, the proposed negotiation strategy effectively assists both agents to reach to an agreement within the specified time-limits.

The rest of the paper is structured as follows. In *Section II*, the negotiation protocol adopted is presented, which does not follow the usual alternating sequential offers pattern, but instead employs a contract ranking mechanism. *Section III* elaborates on the proposed negotiation model, which enhances the existent models introducing the decision issues concept. *Section IV* presents the designed negotiation strategy, which is adequate for cases where the rationale of the BAs is limited. Finally, in *Section V* conclusions are drawn and directions for future plans are presented.

II. A NEGOTIATION PROTOCOL

Any successful negotiation framework imposes the design of an appropriate protocol that will govern the interactions between the negotiators. Depending on the specific negotiation problem that needs to be solved, a protocol comprises a set of rules that constrain the proposals that the negotiation parties are able to make. In [6] a generic framework for automated negotiation is presented. The simplest protocol, which minimises the complexity of the rationale behind the decision models of the agents, specifies that the agents can only accept or reject others' proposals. However, in complex cases where multiple issues are considered, this convention may lead to a time-consuming and inefficient process. In order to improve on the efficiency of the negotiation process, the agents should provide some feedback on the proposal they receive. This feedback may take the form of a critique (e.g., list of comments on elements of the proposal the agent likes or dislikes), or a counter proposal (i.e., an alternative proposal more favorable to its sender, generated in response to an offer). Thus, the probability of an agreement is increased.

In relative research literature, the interactions among the parties follow mostly the rules of an alternating sequential protocol in which the agents in turn make offers and counter offers (e.g., [8]). This model however requires an advanced reasoning component on behalf of the BA as well as the SA. In the context of this paper we tackle the case where the BA does not give a counter offer (which involves incorporating to the model all BA's trade-offs between the various attributes) to the SA, but ranks the SA's offers instead. This ranking is then provided to the SA, in order to generate a better proposal. This process continues until a mutually acceptable contract is reached. This is more efficient in cases in which the BA is not able to extract all user requirements and preferences in a completely quantified way, while being capable of selecting, classifying or rating the contract(s) proposed.

Once the agents have determined the set of issues over which they will negotiate, the negotiation process consists of an alternate succession of contract proposals on behalf of the SA and subsequent ranking of them by the BA according to its preferences and current conditions. Thus, at each round, the SA sends to the BA N contracts (i.e., N packets consisting on n-plets of values of the n contract issues), which are subsequently evaluated by the BA and a rank vector is returned to the SA. This process continues until a contract proposed by the SA is accepted by the BA or one of the agents terminates the negotiation (e.g., if the time deadline is reached without an agreement being in place). Even though negotiation can be initiated by SAs or BAs, only the SAs propose concrete contracts, as there is no counter offer generation mechanism for the BAs. We hereafter consider the case where the negotiation process is initiated by the BA who sends to the SA an initial *Request for Proposal* (*RFP*) specifying the types and nature of the contract issues and the values of all non negotiable parameters. The main issue is assumed to be the price of the good under negotiation, while various other issues may be considered as well. The negotiation protocol adopted has not been described in detail here, as its mathematical formulation is presented in subsequent sections to make the proposed approach more comprehensive to the reader.

III. AN EFFICIENT NEGOTIATION MODEL

In this section, an efficient dynamic negotiation model is presented, based on the multi-issue value scoring system introduced by Raiffa [9], in the context of bilateral negotiations involving a set of quantitative variables. Our aim is to extend this framework into a multi-party, multi-issue, dynamic model. This is important since multilateral negotiations are common in the environment of the electronic marketplace. Based on the designed negotiation protocol, the proposed model is exploited by the SA to create subsequent contracts, while used by the BA to evaluate and rate the contracts offered.

It has been argued in the literature (e.g., [3]) that Raiffa's framework, is based on several implicit assumptions that, even though they may lead to good optimisation results, they are inappropriate for the needs of the e-marketplace, such as: (i) privacy of information for the negotiators is not supported, (ii) the utility function models must be disclosed, (iii) the value regions for the contract issues for both parties must be identified in advance, (iv) the only parameters that determine the utility of the contracts for the negotiators, are the values of the issues under negotiation.

However, there are usually several issues, that even though their values are not under negotiation and they are not included in the contract parameters, they affect the evaluation of the values of the contract issues. Without being exhaustive, such issues may consist of: the number of competitor companies, the number of substitute or complementary products/services, the quantity of product in stock, the number of current potential buyers, the reputation/reliability of each Seller/Buyer, the time until the negotiation deadline expires, the resources availability and restrictions, etc. We will refer to these issues as *decision issues* (*DIs*). The values of the DIs may change overtime, depending on the e-marketplace conditions and on the Seller's and Buyer's state. The DIs not only affect the evaluation of the contracts, but they also have an impact on the generation of subsequent offers. It is noted here that DIs' values do not necessarily depend on the actions of the negotiating party they affect, while they may affect one or both negotiators. The values of the DIs should have a strong and direct influence on the behaviour of the negotiating agents, as they should be able to evaluate the utility of the contracts under the current circumstances in the e-marketplace and act accordingly.

From the above analysis, it is clear that optimal solutions cannot be found in the e-commerce domains, as computational and communication resources usually impose non-zero negotiation duration and time-varying issues may change the conditions for both parties. Thus, we propose a dynamic model for agent negotiation that can be exploited by strategies in order to determine contracts acceptable to all parties, but which, nevertheless, maximise the agent's own utility function.

The agents that represent *Sellers* will be denoted by $S = \{S_1, S_2, ...\}$ and the ones that represent potential *Buyers* will be denoted by $B = \{B_1, B_2, ...\}$. For the values of the DIs we will use the following notation: d_j, $j = 1,...,m$. We may now introduce the utility function of the proposed framework as follows [5]. Let $U_i^a : [m_i^a, M_i^a] \rightarrow [0,1]$ express the utility that agent $a \in S \cup B$ assigns to a value of contract issue i in the range of its acceptable values. Let w_i^a be the importance of issue i for agent a. We assume the weights of all agents are normalised to add up to 1, i.e., $\sum_{i=1}^{n} w_i^a = 1$. Using the above notation, the agent's a utility function for a contract $C_k = \{c_{k1}, ..., c_{kn}\}$ can be defined as follows: $U^a(C_k) = \sum_{i=1}^{n} w_i^a U_i^a\left(c_{ki}, d_j^{t=t_k}\right)$, where $d_j^{t=t_k}$, $j = 1,...,m$, is the value of decision issue d_j at the time t_k, when contract C_k is proposed. Furthermore, it should be noted that the utility function $U_i^a\left(c_{ki}, d_j^{t=t_k}\right)$ may be any functional form (e.g., linear, polynomial, exponential, quasilinear, etc.) of the contract issue c_{ki} value and of the decision issue d_j value at the time contract C_k is proposed, as nonlinear approaches could be used to model the overall utility, without affecting the basic ideas of the model. Examples of utility functions formulations are evaluated in [5].

In order for the utility function of any contract issue i for any negotiator to lie within the range $[0,1]$, the value of issue i must lie within the range of its acceptable values. To ensure this, we introduce the notion of *value constraints*, that is expressed as follows: $m_i^a \leq c_i \leq M_i^a$. In case the *value constraints* hold for all contract issues, the utility function can be used to measure the satisfaction of a negotiator as far as the proposed contract is concerned. Nevertheless, often, the value constraints are not met for some contract issues. In this case, there is not much value in using the above specified utility function to measure the satisfaction degree of this negotiator, as the contract is completely unacceptable. In that sense, agents exhibit lexicographic preferences. Thus, we may introduce a *value constraint validity vector*: $VCV^a = [VCV_i^a]$, $i = 1,...,n$, where $VCV_i^a \in \{0,1\}$, depending on whether the value constraint for negotiating party a is met for contract issue i (i.e., $VCV_i^a = 1$) or not (i.e., $VCV_i^a = 0$)[1].

In principle, SAs and BAs present conflicting interests in the values of the contract issues. Thus, the utility functions must verify that given a Seller agent S and a Buyer agent B negotiating the value for contract issue i, then: $\left[\partial(U_i^S)/\partial c_i\right] \cdot \left[\partial(U_i^B)/\partial c_i\right] < 0$, i.e., under the same conditions, in case higher values of contract issue i result in higher (lower) utility for the SA at the same time they result in lower (higher) utility for the BA. Nevertheless, it must be mentioned that there are cases where the SAs and BAs may have a mutual interest for the value of a contract issue [9].

As already mentioned in *Section II*, the BA ranks the contracts proposed by the SA. For the simplest ranking function, the ranks that may be assigned to any contract proposed are boolean variables, i.e., one instance of the set $\{accept, reject\}$. In a more sophisticated approach, the ranks lie within a range $[m_r, M_r]$, where any contract rated with less than M_r is not acceptable by the BA, while, when a contract is rated with M_r, then the proposed by the SA contract is accepted by the BA. In order to signal the case where at least one value constraint is not met for the BA for a certain contract, we introduce another parameter called *contract value constraints validity* that will be denoted by $CVCV_k^a$ for contract C_k and is given by the following equation: $CVCV_k^a = \prod_{i=1}^{n} VCV_{ki}^a$. Based on the previous analysis, in case all *value constraints* are met for contract C_k, it stands that $CVCV_k^a = 1$. On the other hand, in case at least one *value constraint* is not valid for contract C_k, it stands that $CVCV_k^a = 0$, and then the particular contract is definitely rejected.

In order to introduce the time parameter in our negotiation model, we represent by $P^t = \{C_1^t, ..., C_N^t\}$ the vector of the $N \geq 1$ contracts proposed by the Seller agent S to the Buyer agent B at time t, by $C_k^t = \{c_{k1}^t, ..., c_{kn}^t\}$ the vector of the n contract issues values proposed by S to B at time t for the k-contract of this proposal ($k = 1,...,N$), and by c_{ki}^t ($i = 1,...,n$) the value for issue i proposed by S to B at time t for the k-contract of this proposal. Let now $R^t = \{r_1^t, ..., r_N^t\}$ be the vector of ranking values that B assigns at time t to the previous contracts proposal made by S, and r_k^t ($k = 1,...,N$) be the rank that B assigns at time t to the k-contract of this proposal. The range of values acceptable to agent $a \in \{S, B\}$ for issue i will be represented as the interval $[m_i^a, M_i^a]$.

A contract package proposal is accepted by B when at least one contract is rated with M_r, while the negotiation terminates also in the case where a boolean variable expressing the wish of the agents to quit the negotiation is set to true. The wish of S to quit (continue) the negotiation at time t will be expressed by $Q_S^t = 1$ ($Q_S^t = 0$), while the wish of B to quit (continue) the negotiation at time t will be expressed by $Q_B^t = 1$ ($Q_B^t = 0$). In the first case an agreement is reached and we call the negotiation successful, while in case one of the negotiating parties quits (i.e., its deadline expired) it is called unsuccessful. In any other case, we say that the negotiation thread is active.

IV. The Proposed Negotiation Strategy

Hereafter, our focus is laid on the rationale of the SA, since its adopted strategy will define the outcome of the negotiation, while rather simplified assumptions regarding BA's logic are subsequently made. We consider that a negotiation is successful, if a mutually acceptable contract is reached within reasonable time. Since an exhaustive

[1] In order to refer to the case where the mere presence or absence of a particular feature is required by a negotiator, we could add boolean constraints to our model. However, as they can be reduced to value constrains [5], they will not be further analysed.

exploration of the possible contract space may form a computationally intensive task for the SA, it should be able to infer the acceptable contract space for the BA until a predefined deadline. To be more specific, SAs hereafter will be provided with a mechanism enabling them to find good (near optimal) solutions in reasonable time, by means of computationally efficient algorithms. The rest of this section is structured as follows. In *subsection A* the negotiation problem is described formally, while in *subsection B* an innovative negotiation strategy is thoroughly presented.

A. Negotiation Problem Description

The objective of our problem is to find a contract $C_{final} = \{c_{1final}, c_{2final}, ..., c_{nfinal}\}$ that maximises the SA's overall utility function $U^S(C_{final})$, i.e., the SA's satisfaction stemming from the proposed contract. The constraints of our problem are the following. First, each contract issue-i ($i = 1,...,n$) should lie within the acceptable value ranges for both the BA and the SA, i.e., no value constraint violation should exist for both parties. Second, the constraint regarding the BA's (SA's) utility reservation value should be preserved. Therefore, the total BA's (SA's) utility for a proposed contract should not lie below a predefined value $U^B_{\min Acc}$ ($U^S_{\min Acc}$) representing the minimum satisfaction that may be experienced by the BA (SA) in order for an agreement to be reached. Thus, the conditions $U^B(C_{final}) \geq U^B_{\min Acc}$ and $U^S(C_{final}) \geq U^S_{\min Acc}$ should hold. Finally, the constraint regarding the SA's deadline should be preserved. Therefore, an agreement with the BA may be reached only if $t_l \leq T$, where T denotes the SA's deadline and t_l the time of negotiation round l during which contract C_{final} is proposed.

Thus, based on the selected protocol and the proposed model, designing a negotiation strategy can be reduced to a decision problem that can formally be stated as follows:

Given: (i) two negotiating parties: an SA that may provide a specific good (i.e., service or product) and a BA that is interested in this good's acquisition, (ii) n contract issues (index: $i = 1,...,n$) defined by the negotiators and the acceptable for the SA ranges $\left[m_i^S, M_i^S\right]$ within which their values must lie, (iii) m decision issues and their current values d_j, $j = 1,...,m$, (iv) a deadline T up to which the SA must have completed the negotiation with the BA, (v) the vector $P^{t_l} = \left\{C_1^{t_l},...,C_N^{t_l}\right\}$ of the N contracts $C_k^{t_l} = \left\{c_{k1}^{t_l},...,c_{kn}^{t_l}\right\}$ ($k = 1,...,N$) proposed by the Seller agent S to the Buyer agent B during the previous round l, (vi) the vector $R^{t_l} = \left\{r_1^{t_l},...,r_N^{t_l}\right\}$ of the ranking values $r_k^{t_l}$ ($k = 1,...,N$) that the BA assigns to the previously made by the SA contract proposal at the negotiation round l, and (vii) the value constraint validity vector $VCV_k^B = \left\{VCV_{ki}^B\right\}$ ($i = 1,...,n$) for at least one of the contracts proposed, *find* the vector $P^{t_{l+1}} = \left\{C_1^{t_{l+1}},...,C_N^{t_{l+1}}\right\}$ of the N contracts $C_k^{t_{l+1}} = \left\{c_{k1}^{t_{l+1}},...,c_{kn}^{t_{l+1}}\right\}$ ($k = 1,...,N$) that should be proposed by the SA to the BA in the next round $l+1$, in order to eventually reach to an acceptable (near optimal) agreement between the two parties, while the SA aims to maximise its individual utility of the agreed contract under the SA's constraints, i.e., $VCV_k^S = \left\{VCV_{ki}^S\right\} = \underline{1}$ ($i = 1,...,n$), $U^S(C_k^{t_{l+1}}) \geq U^S_{\min Acc}$ and $t_l \leq T$, and subject to the existent resource and computational limitations.

In general, there may be a significant amount of computations associated with the optimal solution of the negotiation problem presented above. Exhaustive search (i.e., algorithms scanning the entire contract space) should be conducted only in case the solution space is not prohibitively large. The cost of the respective solutions is evaluated and finally, the best solution is maintained. The negotiation problem is *NP-hard*, with regards to the number of the contract issues involved and the range of their acceptable values. In this respect, the design of computationally efficient algorithms that may provide good (near-optimal) solutions in reasonable time is required. Classical methods ([10], [11]) in this respect are simulated annealing, taboo search, genetic or greedy algorithms, hybrid or heuristic techniques, etc. Thus, the SA should be capable of selecting distinct contract points from the acceptable contract space in order to reach to an agreement with the BA within the predefined time limits.

B. Negotiation Strategy

In this section an innovative negotiation strategy that fully exploits the potential of the designed negotiation model is described. This strategy is designed based on the following focal assumptions. First, the SA and the BA will reach to an agreement, only if a contract is found, whose contract issues values lie within the acceptable ranges of both negotiating parties, while their individual utilities are above a minimum acceptable threshold. Thus, for the final agreement contract $C_{final} = \left\{c_{f1},...,c_{fn}\right\}$, the following equations must hold: $U^S\left(C_{final}\right) \geq U^S_{\min Acc}$, $c_{fi} \in \left[m_i^S, M_i^S\right]$ and $U^B\left(C_{final}\right) \geq U^B_{\min Acc}$, $c_{fi} \in \left[m_i^B, M_i^B\right]$, where the pre-defined threshold values $U^S_{\min Acc}$ and $U^B_{\min Acc}$ lie within $[U^S_{\min}, U^S_{\max}]$ and $[U^B_{\min}, U^B_{\max}]$, respectively. Second, it is assumed that the values of all decision issues are invariable and equal to $\underline{d}^{t_0} = \left\{d_j^{t_0}\right\}$ for the maximum possible duration T of the negotiation procedure between the SA and the specific BA, where t_0 is the initiation time of the specific negotiation thread. Third, the duration $t_{l+1} - t_l$ of each negotiation round l is considered to be almost constant. Thus, the maximum number of rounds within which the SA is authorised to complete the negotiation with the BA is: $L = INT(T/(t_{l+1} - t_l))$. Fourth, we assume that the SA sends to the BA only one contract ($N = 1$) at each negotiation round l on the basis of the BA's response to the previous contract proposal. The rest of the section is structured as follows. *Subsection B.1* describes the ranking mechanism of the BA, while *subsection B.2* presents in detail the negotiation strategy designed for the SA.

1) The Ranking Mechanism of the Buyer

Hereafter, we consider the simplest ranking function, according to which the ranks that may be assigned to any contract proposed are boolean variables, i.e., one instance of the set $\{accept, reject\}$. Following the presented notation, a contract can be accepted when all *value constraints* are met (i.e., $\prod_{i=1}^{n} VCV_i^a = 1$) or is definitely rejected when at least one *value constraint* is not valid (i.e., $\prod_{i=1}^{n} VCV_i^a = 0$).

In case of a value constraint violation, at negotiation round l the BA along with a respective identifier (i.e., $r^{t_l} = 0$) sends to the SA the value constraint validity vector VCV^{B,t_l}. As already mentioned an agreement between the negotiating agents is achieved upon the first contract proposed by the SA for which all value and utility constraints hold for the BA as well. In the latter case, the BA sends to the SA an ending sign ($r^{t_l} = 1$), indicating that the specific contract is the final one and the negotiation ends successfully.

Thereafter, it is assumed that the BA adopts a simplified rationale and acts as a hill climber [12] by accepting at each negotiation round l the contract C^{t_l} proposed only if all its value constraints hold (i.e., $VCV^{B,t_l} = \underline{1}$) and the utility acquired $U^B(C^{t_l})$ exceeds the utility stemming from the *last accepted contract*, denoted hereafter as C_{acc} (i.e., $U^B(C^{t_l}) \geq U^B(C_{acc})$). In such a case, the contract C^{t_l} constitutes the *last accepted contract* and C_{acc} is set equal to C^{t_l}. Otherwise, the contract C^{t_l} is rejected. Alternatively, the BA could adopt an annealing mechanism, as presented in the following subsection, in order to escape from potential local optima [10].

2) The Proposed Negotiation Strategy for the Seller

The designed negotiation strategy bears resemblance to the Simulated Annealing technique [10]. Annealing is the physical process in which a crystal is cooled from the liquid to the solid state. Careful cooling brings the crystal in the minimum energy state. In analogy, a simulated annealing algorithm considers each solution of the optimisation problem as a state, the cost of each solution as the energy of the state, and the optimal solution as the minimum energy state. During each phase of an algorithm based on the simulated annealing paradigm, a new solution is generated by minimally altering the currently best solution. If the new solution improves the objective function value (i.e., the difference between the objective function value of the old and the new solution, Δc, is negative) the new solution becomes the currently best solution. Solutions that decrease the objective function value may also be accepted with probability $e^{-(\Delta c/CT)}$. This way local optima are avoided. CT is a control parameter that ends the algorithm when set to zero (i.e., $CT = 0$ when temperature reaches 0). Additionally, the procedure terminates when a significant number of moves have been made without improving the cost function.

The rationale of the new solution generation mechanism is assigned to the SA, while the BA, adopting the annealing or the hill climbing technique, identifies if the new solution proposed by the SA is acceptable or not on the basis of the utility value acquired. The development of a simulated annealing-based procedure means that the following aspects have to be addressed: configuration space, objective function, "neighbourhood" structure and cooling schedule (i.e., manner in which the temperature will be reduced). The configuration space is the set of potentially feasible contracts that satisfy the SA's as well as the BA's related value and utility constraints. The objective function is in essence the utility function of the BA, which is not known to the SA. However, the BA's response will be used in order to determine if the proposed contract was acceptable and generate new potential contract solutions.

As already presented in the negotiation protocol analysis, we consider the case where the negotiation process is initiated by the BA who sends to the SA the initial RFP specifying the types of the contract issues and the values of all non negotiable parameters. Based on this RFP, the SA proposes an initial contract $C^{t_0} = \{c_1^{t_0}, ..., c_n^{t_0}\}$ to the BA at $t = t_0$, setting all contract issues at the values that maximise the SA's utility (i.e., if $\partial[U^S(C_k, \underline{d}^{t_0})]/\partial c_i > 0$, then the SA sets $c_i^{t_0} = M_i^S$, while in case $\partial[U^S(C_k, \underline{d}^{t_0})]/\partial c_i < 0$, then the SA sets $c_i^{t_0} = m_i^S$). The utility of the initial contract C^{t_0} for the SA will be denoted by: $U^S(C^{t_0}, \underline{d}^{t_0}) = U_{\max}^{S,t_0}$, as $U_{\max}^{S,t_0}$ is the maximum utility that can be achieved for the SA, given the values of the decision issues $\underline{d}^{t_0} = \{d_j^{t_0}\}$ at time $t = t_0$.

With respect to the initial contract C^{t_0} two distinct cases may be identified. First, no *value constraint violation* exists and the contract C^{t_0} is ranked by the BA with a specific instance of the set $\{accept, reject\}$. Second, *value constraint violation* occurs, in which case value constraint violation identifier is returned to the SA (i.e., $r^{t_l} = 0$), and the BA provides also its *value constraint validity vector* $VCV^{B,t_0}(C^{t_0})$. In case the initial contract C^{t_0} presents a *value constraint violation*, the SA, as a first step tries to acquire a contract which respects BA's value constraints. We will refer to this step as *negotiation phase I*. To this respect, until a non value constraint violating contract C^{t_l} is acquired, at each negotiation round $l > 1$ the SA makes a compromise (concession), reducing its utility by a certain quantity $\Theta^{t_l} = U^S(C_k^{t_{l-1}}, \underline{d}^{t_0}) - U^S(C_k^{t_l}, \underline{d}^{t_0})$. This quantity Θ^{t_l} can be time dependent, it can depend on the current values of the decision issues [5] (following a Boulware [9], Conceder [2] or Linear scheme), or it may be based on imitative behaviour of the SA [13] (depending on the utility compromise of the BA), etc. Without loss of generality we may assume that Θ^{t_l} is constant, i.e., $\Theta^{t_l} = \Theta^{t_0}$, $\forall l = 1, ..., L$, so that when reaching to its deadline the SA proposes a contract at its utility reservation value (i.e., $U^S(C_k^{t_L}, \underline{d}^{t_0}) = U_{\min Acc}^S$) [5].

A new contract is generated based on the contract $C^{t_{l-1}}$ proposed at negotiation round $l-1$, which in principle has all contract issues values equal to the ones of contract $C^{t_{l-1}}$, except from the value(s) $c_i^{t_{l-1}}$ of contract issue(s) i, for which a constraint violation has occurred, ($VCV_i^{B,t_{l-1}} = 0$). For example, in case contract issue k of contract $C^{t_{l-1}}$ violates the value constraints, the new contract proposal would be $C^{t_l} = \{c_{01}^{t_l}, ..., c_{0(k-1)}^{t_l}, c'^{t_l}_{0k}, c_{0(k+1)}^{t_l}, ..., c_{0n}^{t_l}\}$. The value(s) of contract issue(s) k, $c'^{t_l}_{0k}$, are selected so that the utility of contract C^{t_l} for the SA is equal to: $U^S(C^{t_l}, \underline{d}^{t_0}) = U^S(C^{t_{l-1}}, \underline{d}^{t_0}) - \Theta^{t_0}$. In order to reach to a non violating contract within a reasonable number of negotiation rounds, it is assumed that the concession quantity Θ^{t_0} is shared equally amongst the contract issues whose value is not acceptable to the BA. The exact values of contract issues are determined in accordance with the following: $c'^{t_l}_{0i}$: $U^S(c_{0i}^{t_l}, \underline{d}^{t_0}) - U^S(c'^{t_l}_{0i}, \underline{d}^{t_0}) = \left[\sum_{k=1}^{n} \overline{VCV_k^{B,t_l}}\right]^{-1} \cdot \frac{\Theta^{t_0}}{w_i^S}$. This process continues until a non value constraint violating contract C^{t_l} is acquired.

Thereafter, the algorithm moves to the *negotiation phase II*. In this stage the BA does not provide the value constraint validity vector, but the "accept" or "reject" rank of the proposed contract, and the SA's strategy is modified in order to acquire a mutually acceptable contract within reasonable time. The general idea of the proposed SA's approach is to explore the impact of the reallocation of the utility concession to each one of the contract issues involved in the negotiation process prior to conceding an additional utility quantity Θ^{t_0}. Thus, the neighbourhood structure of a solution in the context of negotiation phase II is produced by reallocating the SA's concession quantity Θ^{t_0} from the current contract issue j to another randomly chosen contract issue j'. In case all contract issues have been assigned with the specific utility concession quantity Θ^{t_0}, while an agreement has not been reached yet, the SA proceeds to an additional concession, whereas the same applies for the following n rounds. The cooling schedule may be calculated according to $T' = r \cdot T$, where T is the temperature and r is usually a number that ranges from 0.95 to 0.99.

The simulated annealing-based algorithm may be described as follows.

Basic Simulated Annealing Algorithm

Step 0. Initialisation. Get an initial contract solution, IS, and an initial temperature value T. The currently best solution (CBS) is IS, i.e., $CBS = IS$, and the current temperature value (CT) is T, i.e., $CT = T$.

Step 1. If $CT = 0$, or if the stop criterion is satisfied, the procedure ends and a transition to *step 6* is performed.

Step 2. If $CVCV^B_{CBS} = 0$, a new solution (NS) is found in accordance with the proposed approach for *negotiation phase I*, or else if $CVCV^B_{CBS} = 1$ the NS is found according to the proposed approach for *negotiation phase II*.

Step 3. The difference of the BA's utilities with respect to the two solutions, CBS and NS is found, i.e., the quantity $\Delta U^B = U^B(CBS) - U^B(NS)$ is computed by the BA.

Step 4. If $\Delta U^B \le 0$ then the new solution is accepted by the BA and becomes the currently best solution, i.e., $CBS = NS$. Otherwise, if $\Delta U^B > 0$, then if $e^{-(\Delta U^B / CT)} > p$ ($p < 1$), the new solution becomes the currently best solution, i.e., $CBS = NS$.

Step 5. The cooling schedule is applied, in order to calculate the new current temperature value CT, and a transition to *step 1* is performed.

Step 6. End.

There are various alternatives for realising the stop criterion mentioned in *step 1*. In our version, the algorithm stops when either the BA sends a successful ending signal to the SA, or the SA has reached its deadline, or the acceptable limit for the values of all the contract issues has been reached, without an agreement being in place. Neighbouring solutions (*step 2*) are selected randomly among all the neighbouring ones of the currently best solution, with the same probability for all neighbours.

It is noted that in case an agreement between BA and SA is feasible (that is there exists at least one contract C^{t_i} for which it stands: $U^S(C^{t_i}) \ge U^S_{\min Acc}$ and $U^B(C^{t_i}) \ge U^B_{\min Acc}$, while the value constraints hold for both agents), our approach succeeds in reaching to an agreement. This is due to the assumption that as its deadline approaches, the SA concedes up to its reservation value $U^S_{\min Acc}$.

VI. Conclusions

This paper presented a multiparty, multi-issue, dynamic negotiation model and an effective strategy, to be exploited by mobile intelligent agents in an e-commerce environment. The efficiency of the proposed framework is due to the fact that the Buyer agent adopts a flexible and light reasoning component, which does not necessitate the explicit statement of all preferences and requirements on behalf of the Buyer in a completely quantified way. A ranking mechanism replaces the counter-offer complicated scheme, while potential decision issues are considered. Thus, it supports an evaluation of the contracts proposed, based not only on the values of the issues under negotiation, but also on the e-marketplace conditions and the negotiator's state. The proposed negotiation strategy is adequate for the simple $\{accept, reject\}$ ranking function. Besides its inherent computational and communication advantages, its efficiency is due to the fact that an agreement between BA and SA is reached in any situation it is feasible, before the predefined deadline expires.

The negotiation framework designed has been adopted by self-interested autonomous agents and has performed well on the generation of subsequent offers and the ranking of the contracts proposed, always converging to a mutually acceptable contract, if any. Initial results indicate that the designed framework produces near optimal results, in case the number of the negotiation issues is quite high. Future plans involve its extensive empirical evaluation against existent models and strategies and against the optimal solution of the negotiation problem.

References

[1] M. He, N. R. Jennings, and H. Leung, "On agent-mediated electronic commerce", *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 4, pp. 985-1003, 2003.

[2] D.G. Pruitt, *Negotiation Behavior*. Academic Press Inc., 1981.

[3] P. Faratin, C. Sierra, and N. R. Jennings, "Negotiation Decision Functions for Autonomous Agents", *Int. Journal of Robotics and Autonomous Systems*, vol. 24, no. 3-4, pp. 159-182, 1998.

[4] J. S. Rosenschein and G. Zlotkin, *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers*. Massachusetts: The MIT Press, Cambridge, MA, USA, 1994.

[5] I. Roussaki and M. Louta, "Efficient Negotiation Framework and Strategies for the Next Generation Electronic Marketplace", MBA Thesis, October 2003.

[6] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge, "Automated Negotiation: Prospects, Methods, and Challenges", *Int. Journal of Group Decision and Negotiation*, vol. 10, no. 2, pp. 199-215, 2001.

[7] V. Conitzer and T. Sandholm, "Computational criticisms of the revelation principle", *5th Workshop on Agent Mediated Electronic Commerce*, Melbourne, Australia, 2003.

[8] A. Rubinstein, "Perfect equilibrium in a bargaining model", *Econometrica*, vol. 50, pp. 97-109, 1982.

[9] H. Raiffa, *The Art and Science of Negotiation*. Harvard University Press, Cambridge, USA, 1982.

[10] E. Aarts and J. Korts, *Simulated annealing and the Boltzmann machines*. Publisher: J. Wiley & Sons, New York, 1989.

[11] C. Papadimitriou, and K. Steiglitz, *Combinatorial optimization: Algorithms and complexity*. Prentice Hall, Inc., 1982.

[12] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam, "Negotiating Complex Contracts", *Int. Conf. on Autonomous Agents and Multi-Agent Systems*, Bologna, Italy, vol. 2, pp. 753-757, ACM, 2002.

[13] R. Axeldrod, *The Complexity of Cooperation*. Princeton University Press, USA, 1997.